MFB per Modicon M340 usando Unity Pro

Guida all'avvio

07/2012

35013567.06

www.schneider-electric.com

Questa documentazione contiene la descrizione generale e/o le caratteristiche tecniche dei prodotti qui contenuti. Questa documentazione non è destinata e non deve essere utilizzata per determinare l'adeguatezza o l'affidabilità di questi prodotti relativamente alle specifiche applicazioni dell'utente. Ogni utente o specialista di integrazione deve condurre le proprie analisi complete e appropriate del rischio, effettuare la valutazione e il test dei prodotti in relazioni all'uso o all'applicazione specifica. Né Schneider Electric né qualunque associata o filiale deve essere tenuta responsabile o perseguibile per il cattivo uso delle informazioni ivi contenute. Gli utenti possono inviarci commenti e suggerimenti per migliorare o correggere questa pubblicazione.

È vietata la riproduzione totale o parziale del presente documento in qualunque forma o con qualunque mezzo, elettronico o meccanico, inclusa la fotocopiatura, senza esplicito consenso scritto di Schneider Electric.

Durante l'installazione e l'uso di questo prodotto è necessario rispettare tutte le normative locali, nazionali o internazionali in materia di sicurezza. Per motivi di sicurezza e per assicurare la conformità ai dati di sistema documentati, la riparazione dei componenti deve essere effettuata solo dal costruttore.

Quando i dispositivi sono utilizzati per applicazioni con requisiti tecnici di sicurezza, occorre seguire le istruzioni più rilevanti.

Un utilizzo non corretto del software Schneider Electric (o di altro software approvato) con prodotti hardware Schneider Electric può costituire un rischio per l'incolumità personale o provocare danni alle apparecchiature.

La mancata osservanza di queste indicazioni può costituire un rischio per l'incolumità personale o provocare danni alle apparecchiature.

© 2012 Schneider Electric. Tutti i diritti riservati.

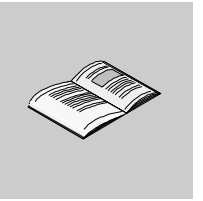
Struttura della documentazione

Documenti correlati

È possibile consultare la seguente documentazione correlata:

- Guida in linea di Unity Pro
- Libreria MFB sulla Guida in linea di Unity Pro
- CD di documentazione **Lexium 15** fornito con il prodotto
- CD di documentazione **Lexium 05** fornito con il prodotto
- Guida in linea di Unilink L per **Lexium 15LP** e di Unilink MH per **Lexium 15MP/HP**
- Guida in linea di PowerSuite for **ATV**
- Guida in linea di PowerSuite for **Lexium 05**
- Guida in linea di Lexium CT for **Lexium 32**
- Guida in linea di SoMove per **ATV 32**

Indice



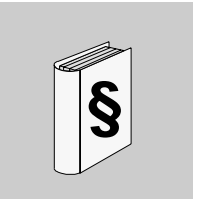
Informazioni di sicurezza 9
Informazioni su... 11
Parte I Guida all'avvio per un'applicazione ad asse singolo 13
Capitolo 1 Foreword 15
Generale 16
Disponibilità dei blocchi sui diversi servoazionamenti 17
Metodologia 19
Capitolo 2 Configurazione applicazione 21
2.1 Ambiente hardware e software 22
Architettura di un'applicazione con un Lexium 05 23
Requisiti software 24
Requisiti hardware 25
2.2 Configurazione dell'applicazione utilizzando Unity Pro 26
Creazione del progetto 27
Configurazione del task Master 28
2.3 Configurazione del bus CANopen 29
Metodologia di implementazione del bus CANopen 30
Configurazione della porta CANopen 31
Configurazione dello slave CANopen 32
Verifica della configurazione del bus CANopen 35
2.4 Configurazione dell'asse usando Motion Tree Manager 36
Directory Motion 37
Creazione e configurazione dell'asse 39
Variabili Axis_Ref, Can_Handler, AxisParamDesc e Recipe 44
Risultato della configurazione della directory Motion 46
2.5 Configurazione del Lexium 05 48
Configurazione di Lexium 05 in PowerSuite 49
Configurazione con l'interfaccia utente di Lexium 05 52
Capitolo 3 Programmazione applicazione 55
Dichiarazione delle variabili 56
Programmazione dell'esempio 57
Blocco funzione CAN_HANDLER 59
Gestione delle modalità di funzionamento e di arresto degli assi 62

	Comando di movimento	63
	Monitoraggio del movimento	65
	Stato e codici di errore degli assi	66
	Backup e trasferimento parametri del servoazionamento	68
	Trasferimento del progetto tra il terminale e il PLC	69
Capitolo 4	**Debug dell'applicazione**	**71**
	Regolazione del Lexium 05 con PowerSuite	72
	Uso dei dati con le tabelle di animazione	73
	Debug del programma	75
	Uso dei dati con le schermate operatore	77
Capitolo 5	**Uso dell'applicazione**	**79**
	Gestione delle ricette	79
Capitolo 6	**Manutenzione dell'applicazione**	**81**
	Esempio di errore	82
	Sostituzione di un servoazionamento guasto	84
Parte II	**Applicazione a più assi**	**87**
Capitolo 7	**Premessa**	**89**
	Architettura dell'applicazione con tutti i servoazionamenti	89
Capitolo 8	**Compatibilità delle applicazioni di movimento con le versioni di Unity Pro**	**91**
		91
Capitolo 9	**Implementazione di Lexium 32 per i Blocchi funzione di movimento**	**93**
9.1	Adattamento dell'applicazione al Lexium 32	94
	Architettura dell'applicazione con Lexium 32	95
	Requisiti software	96
	Requisiti hardware	97
	Configurazione bus CANopen del Lexium 32	98
9.2	Configurazione del Lexium 32	102
	Parametri di base per Lexium 32 con Lexium CT	102
9.3	Regolazione del Lexium 32	105
	Regolazione del Lexium 32	106
	Debug del Lexium 32	107
Capitolo 10	**Lexium 15MP/HP/LP Implementazione dei Blocchi funzione di movimento**	**111**
10.1	Adeguamento dell'applicazione agli azionamenti Lexium 15MP/HP/LP	112
	Architettura di un'applicazione con gli azionamenti Lexium 15MP/HP/LP	113
	Requisiti software	114
	Requisiti hardware	116
10.2	Configurazione del bus CANopen per gli azionamenti Lexium 15MP/HP/LP	117
	Configurazione dello slave CANopen sul bus CANopen	117

10.3 configurazione del Lexium 15MP/HP/LP 120
Parametri di base per Lexium 15MP tramite Unilink MH. 121
Parametri di base per Lexium 15LP tramite Unilink L 124
Parametri specifici per Lexium 15 MP/HP/LP con Unilink. 129
10.4 regolazione del Lexium 15MP/HP/LP 131
Debug dell'asse. 131
Capitolo 11 ATV 31 Implementazione Blocchi funzione di movimento. 135
11.1 Adattamento dell'applicazione all'ATV 31 136
Architettura di un'applicazione con un ATV 31 137
Requisiti software 138
Requisiti hardware. 139
11.2 Configurazione del bus CANopen ATV 31 140
Configurazione dello slave CANopen (ATV 31) sul bus CANopen 140
11.3 Configurazione del ATV 31 143
Configurazione di ATV 31 in PowerSuite 144
Configurazione di ATV 31 con l'interfaccia utente. 147
11.4 Regolazione dell'ATV 31 149
Regolazione dell'ATV 31 con PowerSuite. 149
Capitolo 12 ATV 32 - Implementazione dei Blocchi funzione di movimento (Motion) 151
12.1 Adattamento dell'applicazione all'ATV 32 152
Architettura di un'applicazione con un ATV 32 153
Requisiti software 154
Requisiti hardware. 155
12.2 Configurazione del bus CANopen ATV 32 156
Configurazione dello slave CANopen (ATV 32) sul bus CANopen 156
12.3 Configurazione dell'ATV 32. 159
Configurazione del servoazionamento ATV 32 con SoMove 160
Configurazione dell'ATV 32 con l'interfaccia utente 162
Capitolo 13 ATV 71 Implementazione dei Blocchi funzione di movimento. 165
13.1 Adattamento dell'applicazione all'ATV 71 166
Architettura di un'applicazione con un ATV 71 167
Requisiti software 168
Requisiti hardware. 169
13.2 Configurazione del bus CANopen ATV 71 170
Configurazione dello slave CANopen (ATV 71) sul bus CANopen 170
13.3 Configurazione dell'ATV 71. 173
Configurazione di ATV 71 in PowerSuite 174
Configurazione di ATV 71 con l'interfaccia utente. 177
13.4 Regolazione dell'ATV 71. 179
Regolazione dell'ATV 71 con PowerSuite. 179

Capitolo 14 IclA Implementazione dei Blocchi funzione di movimento 181
14.1 Adattamento dell'applicazione all'IclA 182
Architettura di un'applicazione con un IclA 183
Requisiti software. 184
Requisiti hardware 185
14.2 Configurazione del bus CANopen IclA. 186
Configurazione dello slave CANopen (IclA) sul bus CANopen 186
14.3 Configurazione dell'IclA 189
Configurazione dell'IclA con i microinterruttori DIP 189
14.4 Regolazione dell'IclA 190
Configurazione dell'IclA in IclA Easy 191
Regolazione dell'IclA con IclA Easy. 194
Indice analitico 197

Informazioni di sicurezza



Informazioni importanti

AVVISO

Leggere attentamente queste istruzioni e osservare l'apparecchiatura per familiarizzare con i suoi componenti prima di procedere ad attività di installazione, uso o manutenzione. I seguenti messaggi speciali possono comparire in diverse parti della documentazione oppure sull'apparecchiatura per segnalare rischi o per richiamare l'attenzione su informazioni che chiariscono o semplificano una procedura.

L'aggiunta di questo simbolo a un'etichetta di Pericolo relativa alla sicurezza indica che esiste un rischio da shock elettrico che può causare lesioni personali se non vengono rispettate le istruzioni.

Questo simbolo indica un possibile pericolo. È utilizzato per segnalare all'utente potenziali rischi di lesioni personali. Rispettare i messaggi di sicurezza evidenziati da questo simbolo per evitare da lesioni o rischi all'incolumità personale.

⚠ PERICOLO

PERICOLO indica una condizione immediata di pericolo, la quale, se non evitata, **può causare** seri rischi all'incolumità personale o gravi lesioni.

⚠ ATTENZIONE

ATTENZIONE indica una situazione di potenziale rischio che, se non evitata, **può provocare** morte o gravi infortuni.

⚠ ATTENZIONE

ATTENZIONE indica una situazione di potenziale rischio, che, se non evitata, **può provocare** infortuni di lieve entità.

AVVISO

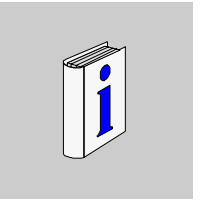
Un ***AVVISO*** è utilizzato per affrontare delle prassi non connesse all'incolumità personale.

NOTA

Manutenzione, riparazione, installazione e uso delle apparecchiature elettriche si devono affidare solo a personale qualificato. Schneider Electric non si assume alcuna responsabilità per qualsiasi conseguenza derivante dall'uso di questi prodotti.

Il personale qualificato è in possesso di capacità e conoscenze specifiche sulla costruzione, il funzionamento e l'installazione di apparecchiature elettriche ed è addestrato sui criteri di sicurezza da rispettare per poter riconoscere ed evitare le condizioni a rischio.

Informazioni su...



In breve

Scopo del documento

Questo manuale descrive, sulla base di un esempio documentato, le modalità d'uso dei blocchi funzione di movimento (MFB) con Modicon M340 con Unity Pro. I blocchi MFB consentono la gestione semplificata dei servoazionamenti e dei servoamplificatori tramite il bus CANopen.

Per poter utilizzare i blocchi MFB, è necessaria una conoscenza approfondita del software Unity Pro, dal momento che la relativa implementazione richiede l'uso delle funzioni standard di questo software (editor dati, IODDT, ecc.).

Inoltre è consigliabile disporre di una conoscenza approfondita del settore specifico del controllo di movimento prima di progettare e implementare un'applicazione di gestione del movimento di assi.

Nota di validità

Questa documentazione è valida dalla versione Unity Pro v7.0

Commenti utente

Inviare eventuali commenti all'indirzzo e-mail techcomm@schneider-electric.com.

Guida all’avvio per un’applicazione ad asse singolo

I

Contenuto della sezione

Questa sezione presenta, sotto forma di esercitazione, un esempio di applicazione di controllo del movimento che implementa i blocchi MFB utilizzando Unity Pro.

Contenuto di questa parte

Questa parte contiene i seguenti capitoli:

Capitolo	Titolo del capitolo	Pagina
1	Foreword	15
2	Configurazione applicazione	21
3	Programmazione applicazione	55
4	Debug dell’applicazione	71
5	Uso dell’applicazione	79
6	Manutenzione dell’applicazione	81

Foreword

1

Argomento del capitolo

Questo capitolo presenta le specifiche dell'applicazione e la metodologia utilizzata nello sviluppo del programma.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Generale	16
Disponibilità dei blocchi sui diversi servoazionamenti	17
Metodologia	19

Generale

Introduzione

L'offerta degli MFB usando Unity Pro rappresenta una nuova funzionalità del controllo di movimento. Tramite il bus CANopen, questa funzionalità permette un accesso semplificato alle funzioni di base sui servoazionamenti e motori a velocità variabile (VSD).

Questa funzionalità, alla quale è possibile accedere tramite il browser del progetto, permette di:

- dichiarare e configurare gli assi in Unity Pro
- creare le variabili del controllo di movimento
- controllare gli assi usando i blocchi funzione elementari del controllo di movimento.

Specifiche

Lo scopo dell'applicazione proposta è di:

- gestire le modalità di funzionamento di un asse lineare usando un tipo di azionamento **Lexium 05**.
- spostare l'asse alla posizione del punto di origine (home), eseguire movimenti reversibili o spostare l'asse a varie posizioni
- dare la possibilità di interrompere il movimento in corso con il comando Stop.

Devono essere prese tutte le disposizioni per eseguire la diagnostica degli errori e il riconoscimento.

Standard

I blocchi della libreria MFB sono conformi con:

- Standard PLCopen

Disponibilità dei blocchi sui diversi servoazionamenti

Blocchi funzione Motion

Non tutti i blocchi sono disponibili su ogni piattaforma hardware. I blocchi disponibili sulla piattaforma Modicon M340 con bus di campo CANopen sono riportati nelle tabelle seguenti.

Tipo	Nome del blocco	ATV 31 ATV312 (7.)	ATV 32	ATV 71	Lexium 32	Lexium 05	Lexium 15 HP, MP, LP	IclA IFA, IFE, IFS
PLCopen motioncontrol V1.1	MC_ReadParameter	X	X	X	X	X	X	X
	MC_WriteParameter	X	X	X	X	X	X	X
	MC_ReadActualPosition				X	X	X	X
	MC_ReadActualVelocity (1.)	X	X	X	X	X	X	X
	MC_Reset	X	X	X	X	X	X	X
	MC_Stop	X	X	X	X	X	X	X
	MC_Power	X	X	X	X	X	X	X
	MC_MoveAbsolute				X	X	X	X
	MC_MoveRelative				X	X	X	
	MC_MoveAdditive				X	X		X
	MC_Home				X	X	X	X
	MC_MoveVelocity	X	X	X	X	X	X	X
	MC_ReadAxisError	X	X	X	X	X	X	X
	MC_ReadStatus	X	X	X	X	X	X	X
	MC_TorqueControl (1.)			X	X	X	X (3.)	
	MC_ReadActualTorque (1.)	X	X	X	X	X	X	
	MC_Jog (2.)				X	X	X (3.), tranne 15 LP	X
Funzioni di salvataggio e ripristino delle serie di parametri per la gestione delle ricette o la sostituzione dei servoazionamenti guasti	TE_UploadDriveParam	X	X	X	X (6.)	X	X	X
	TE_DownloadDriveParam	X	X	X	X (6.)	X	X	X

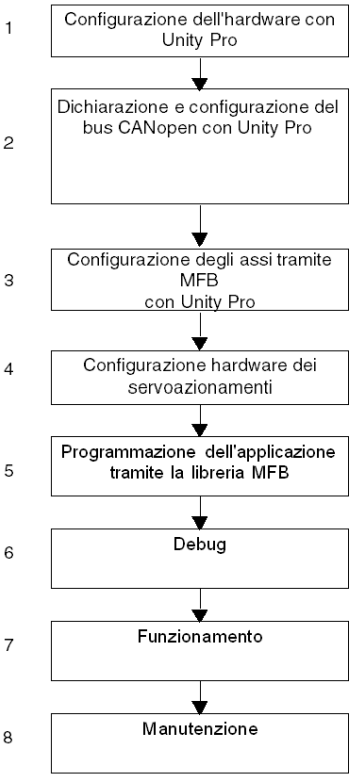
Tipo	**Nome del blocco**	**ATV 31 ATV312 (7.)**	**ATV 32**	**ATV 71**	**Lexium 32**	**Lexium 05**	**Lexium 15 HP, MP, LP**	**IclA IFA, IFE, IFS**
Funzioni avanzate per **Lexium**	Lxm_GearPos					X (4.)	X (5.)	
	Lxm_GearPosS				X	X (4.)	X (5.)	
	Lxm_UploadMTask						X	
	Lxm_DownloadMTask						X	
	Lxm_StartMTask				X		X	
Funzione di sistema	CAN_Handler	X	X	X	X	X	X	X

1. PLCopen V0.99 parte di estensione 2
2. Non PLCopen standard
3. Solo per versione firmware >= 6.73
4. Solo per versione firmware >= 1.403
5. Solo per versione firmware >= 2.36
6. L'elenco dei parametri è un elenco parametri dell'azionamento Lexium32Advanced
7. Attraverso una configurazione del dispositivo ATV 31 V1.7 CANopen.

Metodologia

Panoramica

Il seguente diagramma di flusso illustra le varie fasi dell'installazione di un'applicazione:



La tabella che segue contiene una descrizione dettagliata dei task da eseguire in ogni fase del diagramma di flusso:

Passo	Descrizione
1	In Unity Pro: ● creazione del progetto e selezione del processore
2	In Unity Pro: ● apertura di una configurazione del bus CANopen ● selezione dello slave CANopen nel catalogo hardware ● assegnazione di un indirizzo topologico al nuovo dispositivo ● verifica o impostazione della funzione MFB nella finestra di configurazione del dispositivo ● attivazione della configurazione CANopen ● verifica dell'accuratezza della configurazione tramite la struttura ad albero CANopen nel browser del progetto.
3	Creazione degli assi nella directory **Motion** del browser del progetto. Definizione delle variabili associate agli assi in fase di creazione.
4	Con il software PowerSuite: ● collegamento al dispositivo ● immissione dei parametri richiesti per il funzionamento corretto della comunicazione CANopen (indirizzo, velocità, ecc.).
5	Programmazione delle sequenze di movimento con i blocchi funzione appropriati della libreria MFB. Associazione delle variabili definite durante la creazione degli assi con i blocchi MFB.
6	Debug degli assi con PowerSuite. In Unity: ● debug del programma tramite la tabella di animazione ● uso dei dati tramite le schermate operatore
7	gestione delle ricette di produzione con i blocchi funzione appropriati della libreria MFB: ● creazione e backup delle ricette ● trasferimento dei dati dalle ricette.
8	Backup dei dati e procedure di ripristino.

Configurazione applicazione

2

Argomento del capitolo

In questo capitolo vengono descritte le diverse fasi necessarie alla configurazione dell'applicazione.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
2.1	Ambiente hardware e software	22
2.2	Configurazione dell'applicazione utilizzando Unity Pro	26
2.3	Configurazione del bus CANopen	29
2.4	Configurazione dell'asse usando Motion Tree Manager	36
2.5	Configurazione del Lexium 05	48

2.1 Ambiente hardware e software

Argomento della sezione

Questa sotto sezione descrive gli ambienti hardware e software utilizzati nell'applicazione.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Architettura di un'applicazione con un Lexium 05	23
Requisiti software	24
Requisiti hardware	25

Architettura di un’applicazione con un Lexium 05

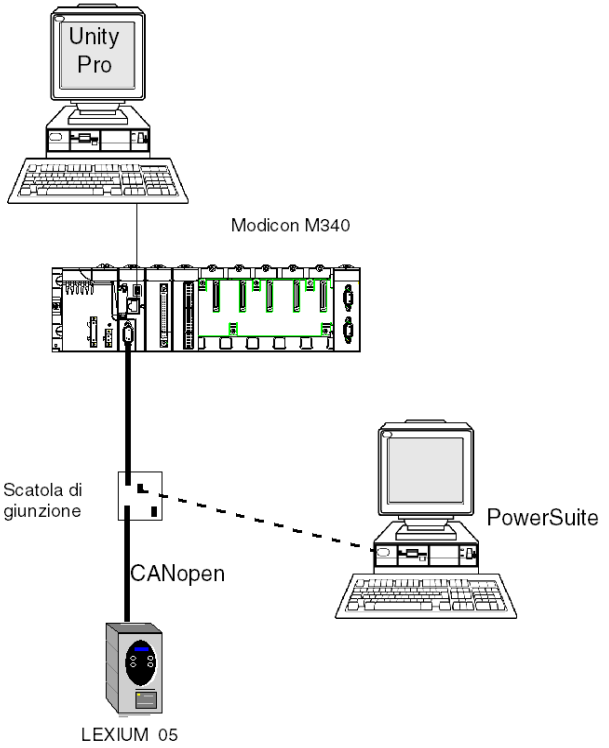
Panoramica

L’architettura descritta è semplice e integra i principi di implementazione del controllo di movimento.

A questo tipo di architettura possono essere aggiunti altri dispositivi per consentire la gestione di più assi.

Illustrazione

Nella seguente figura è illustrata l’architettura utilizzata nell’applicazione che comprende un **Lexium 05**.



Requisiti software

Panoramica

Per implementare l'esempio, è indispensabile disporre di determinati elementi software su un singolo PC. In particolare, questo consente la configurazione, l'impostazione dei parametri e l'azionamento dei vari dispositivi utilizzati.

L'architettura software è composta dai seguenti elementi:

- Unity Pro, utilizzato per controllare il servoazionamento tramite il bus CANopen programmando i movimenti
- Powersuite, utilizzato per impostare i parametri e regolare il servoazionamento Lexium 05

È tuttavia possibile fare a meno di PowerSuite in alcuni casi, utilizzando l'interfaccia utente *(vedi pagina 52)* del pannello frontale del **Lexium 05**.

Versioni

Nella seguente tabella sono elencate le versioni hardware e software utilizzate nell'architettura *(vedi pagina 23)* che consentono 'uso degli MFB in Unity Pro.

Hardware	**Versione software utilizzata nell'esempio**	**Versione firmware**
Modicon M340	Unity Pro V5.0	-
Lexium 05	PowerSuite for **Unity V5.0** V2.5, patch V2.2.0B	V1.403

Requisiti hardware

Codici prodotto dei componenti hardware utilizzati

Nella seguente tabella sono elencati i componenti hardware utilizzati nell'architettura *(vedi pagina 23)* che permettono l'implementazione degli MFB **Lexium 05** in Unity Pro.

Hardware	**Riferimento**
PLC **Modicon M340**	**BMX P34 2030**
Alimentatore **Modicon M340**	**BMX CPS 2000**
Rack **Modicon M340**	**BMX XBP 0800**
Scatola di giunzione CANopen tra **Modicon M340** e il servoazionamento **Lexium 05**	**VW3CANTAP2**
Cavo di programmazione RJ45 con adattatore RS485/RS232 tra la scatola di giunzione e il servoazionamento	**ACC2CRAAEF030**
Servoazionamento **Lexium 05**	**LXM05AD10M2**
Motore **Lexium 05**	**BSH0551T**

NOTA: Il resistore di terminazione è integrato nel **Lexium 05**.

2.2 Configurazione dell’applicazione utilizzando Unity Pro

Argomento della sezione

Questa sotto sezione descrive la configurazione hardware utilizzando Unity Pro.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Creazione del progetto	27
Configurazione del task Master	28

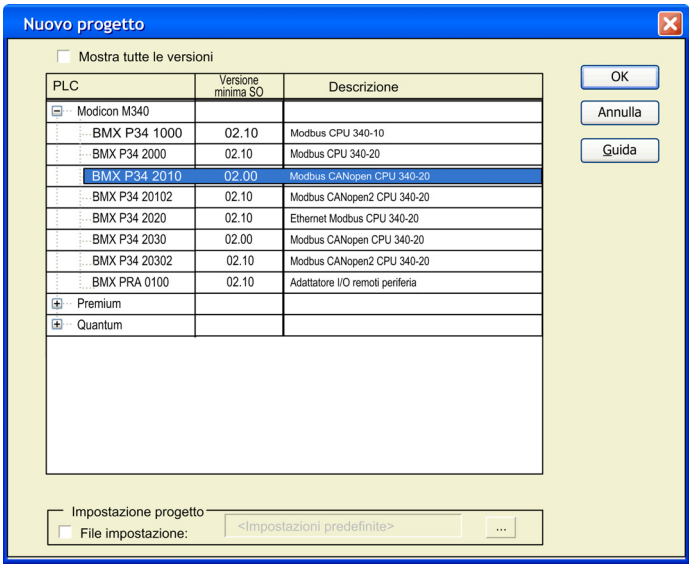
Creazione del progetto

In breve

Lo sviluppo di un'applicazione mediante Unity Pro consiste nella creazione di un progetto associato con un PLC.

Procedura per la creazione di un progetto

La tabella seguente descrive la procedura di creazione di un progetto mediante Unity Pro.

Passo	Azione
1	Avviare il software Unity Pro.
2	Fare clic su **File** poi su **Nuovo** e infine selezionare un PLC 
3	Per vedere tutte le versioni del PLC, fare clic sulla casella Mostra tutte le versioni.
4	Selezionare il processore da utilizzare tra quelli proposti.
5	Per creare un progetto con valori specifici di impostazione, selezionare la casella **File impostazione** e usare il pulsante del browser per identificare il file .XSO (File impostazione progetto). È anche possibile crearne uno nuovo. Se non viene selezionata la casella **File impostazioni**, sono utilizzati i valori predefiniti delle impostazioni di progetto.
6	Confermare facendo clic su **OK**. Per impostazione predefinita, l'applicazione inserisce un rack e un alimentatore.

Configurazione del task Master

Informazioni generali

La prima operazione da effettuare per creare un programma d'applicazione è quella di definire i **Task**.

Viene consigliato di programmare il servoazionamento utilizzando i blocchi MFB nel **MAST**. Questo task deve essere analizzato ad intervalli regolari.

⚠ ATTENZIONE

FUNZIONAMENTO ANOMALO DEI BLOCCHI MFB

Non mischiare task MAST con i task FAST. E' possibile utilizzare i task FAST per programmare i blocchi MFB.

Il mancato rispetto di queste istruzioni può provocare infortuni o danni alle apparecchiature.

Configurazione

La seguente tabella descrive la procedura d'impostazione dei parametri del task **MAST**:

Passo	Azione
1	Nel **Browser del progetto** espandere la directory **Programma**. Viene visualizzata la directory **MAST**.
2	Fare clic con il pulsante destro sulla directory **MAST**, ed eseguire il comando **Proprietà** dal menu contestuale.
3	Fare clic su **Proprietà** per visualizzare la seguente finestra di dialogo:
4	Selezionare il tipo di scansione **Periodica**.
5	Impostare la durata del task a 20.
6	Impostare il valore del **Watchdog** con un valore maggiore del valore della durata.
7	Fare clic su **OK** per confermare la configurazione.

2.3 Configurazione del bus CANopen

Argomento della sezione

Questa sezione presenta la metodologia per la configurazione del bus CANopen.

Contenuto di questa sezione

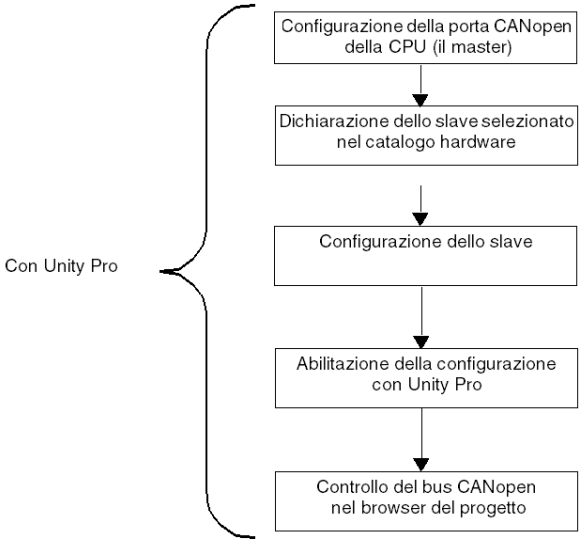
Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Metodologia di implementazione del bus CANopen	30
Configurazione della porta CANopen	31
Configurazione dello slave CANopen	32
Verifica della configurazione del bus CANopen	35

Metodologia di implementazione del bus CANopen

Panoramica

Il diagramma di flusso seguente mostra la metodologia di implementazione di un bus CANopen con Modicon M340.


Con Unity Pro
Configurazione della porta CANopen della CPU (il master)
Dichiarazione dello slave selezionato nel catalogo hardware
Configurazione dello slave
Abilitazione della configurazione con Unity Pro
Controllo del bus CANopen nel browser del progetto

Configurazione della porta CANopen

In breve

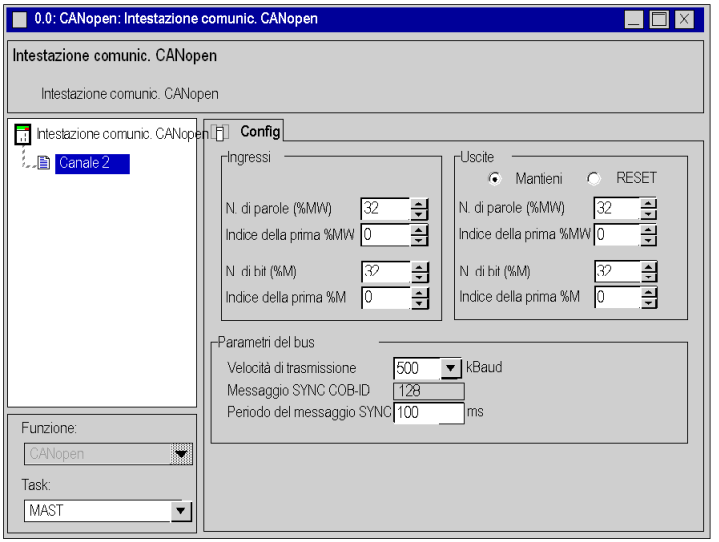
Con Unity Pro è possibile definire il bus CANopen.

Il master del bus CANopen è una porta integrata nella CPU.

Innanzitutto occorre configurare il master del bus.

Come configurare il master del bus CANopen

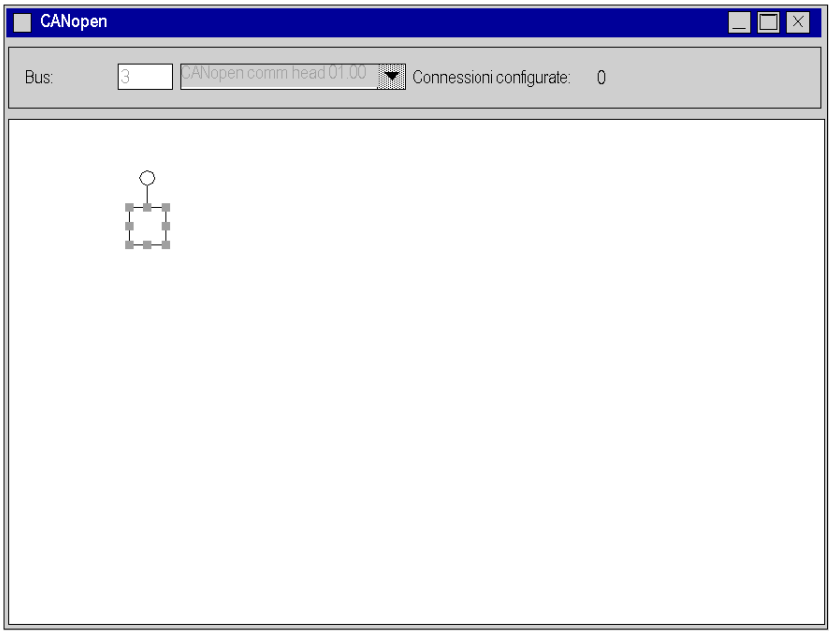
In questa tabella è descritta la procedura di configurazione della porta CANopen con Unity Pro.

Passo	Azione
1	Nel **Browser del progetto** di Unity Pro, espandere completamente la directory **Configurazione**, quindi fare doppio clic sul bus del PLC.
2	Fare doppio clic sulla porta CANopen del PLC. **Risultato:** viene visualizzata la finestra di configurazione della porta: 
3	Nella zona **Parametri del bus**, impostare 500 kBaud come **velocità di trasmissione**. Nella zona **Task**, selezionare **MAST**. Nell'area **Uscite** selezionare il pulsante di selezione **Reset**. (Molto raccomandato)
4	Confermare la configurazione.
5	**Nota:** si raccomanda di utilizzare l'IODDT T_COM_CO_BMX corrispondente alla porta CANopen per la parte restante della programmazione. Immettere CAN come **prefisso nome**. Chiudere la finestra.

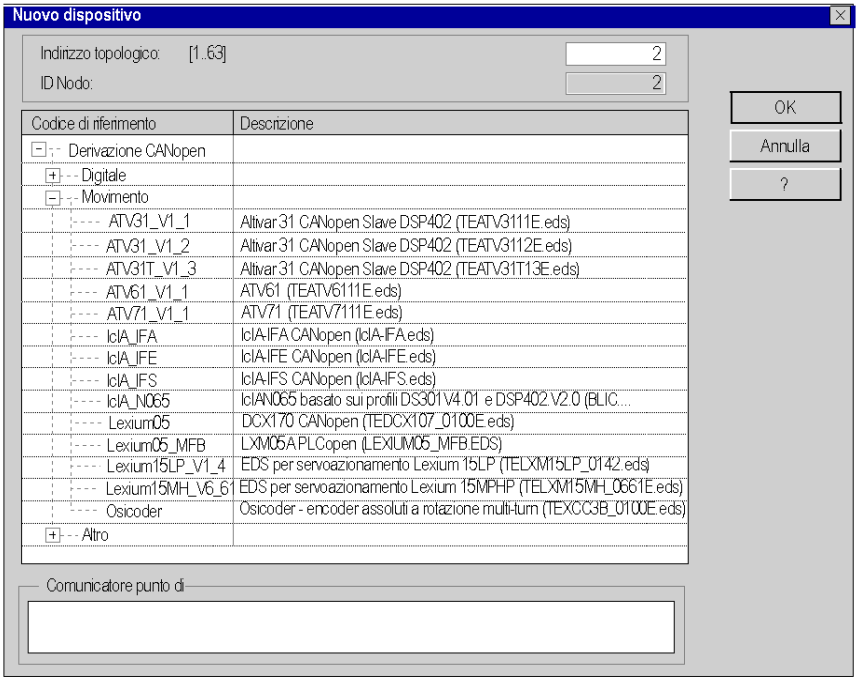
Configurazione dello slave CANopen

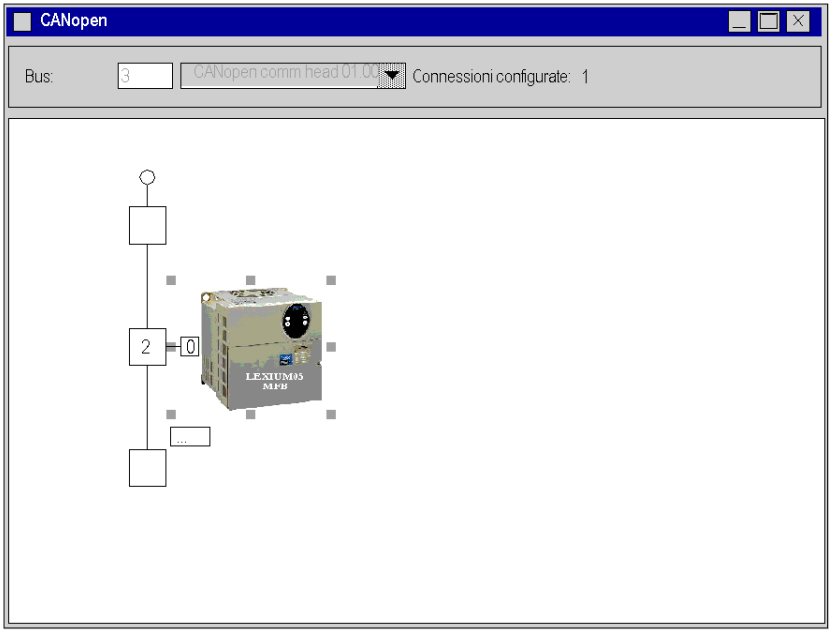
Configurazione dello slave CANopen

Questa tabella descrive la procedura per configurare lo slave CANopen.

Passo	Azione
1	Nel **Browser del progetto** di Unity Pro, espandere completamente la directory **Configurazione**, quindi fare doppio clic su **CANopen**. **Risultato:** viene visualizzata la finestra CANopen: 

Passo	Azione
2	Selezionare **Modifica** →**Nuova apparecchiatura**. **Risultato:** viene visualizzata la finestra Nuova apparecchiatura:
3	Impostare 2 in Indirizzo topologico: Scegliere Lexium05_MFB come dispositivo slave.



Passo	Azione
4	Fare clic su OK per confermare la scelta. **Risultato:** viene visualizzata la finestra CANopen con la nuova apparecchiatura selezionata: 
5	Selezionare **Modifica →Apri modulo**. Se MFB non è già selezionato, selezionarlo nell'area Funzione.
6	Selezionare la scheda **Controllo errore**. Verificare che il valore **Durata generatore impulsi Heartbeat del nodo** sia uguale a 300ms.
7	Il sistema chiede di convalidare le modifiche alla chiusura delle finestre Apparecchiatura e CANopen.

Verifica della configurazione del bus CANopen

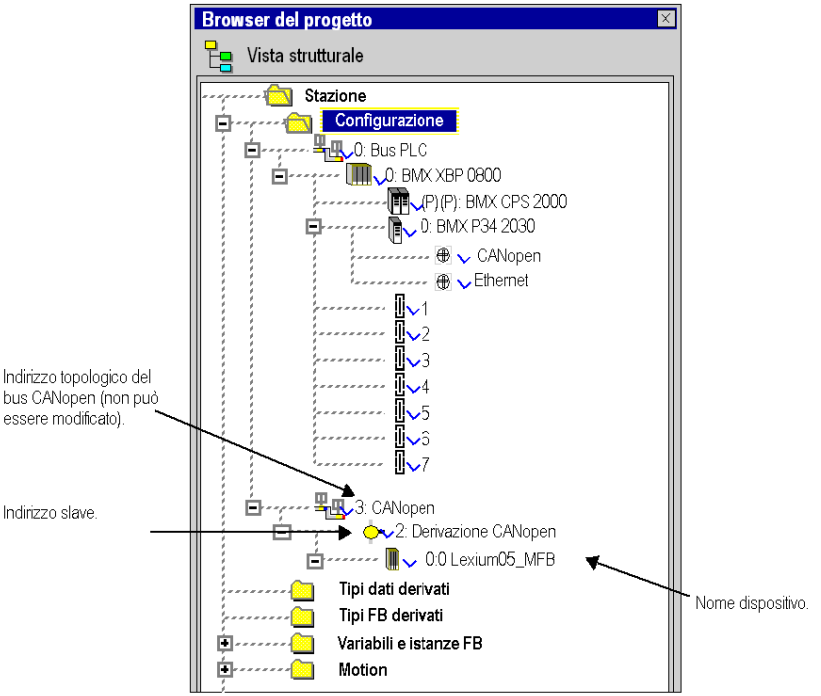
In breve

Il bus CANopen è rappresentato nella directory **Configurazione** del browser del progetto.

Dopo aver selezionato e abilitato la configurazione CANopen, gli slave CANopen vengono visualizzati nel **Browser del progetto**.

L'indirizzo topologico del bus CANopen è calcolato automaticamente da Unity Pro. Questo valore non può essere modificato.

Nel seguente schema è illustrato il bus CANopen con il dispositivo slave indicati nell'esempio di esercitazione:



2.4 Configurazione dell’asse usando Motion Tree Manager

Argomento della sezione

Questa sotto sezione descrive la directory **Motion** aggiunta al browser del progetto di Unity Pro. Presenta inoltre una procedura per la creazione dell’asse in questa directory.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Directory Motion	37
Creazione e configurazione dell’asse	39
Variabili Axis_Ref, Can_Handler, AxisParamDesc e Recipe	44
Risultato della configurazione della directory Motion	46

Directory Motion

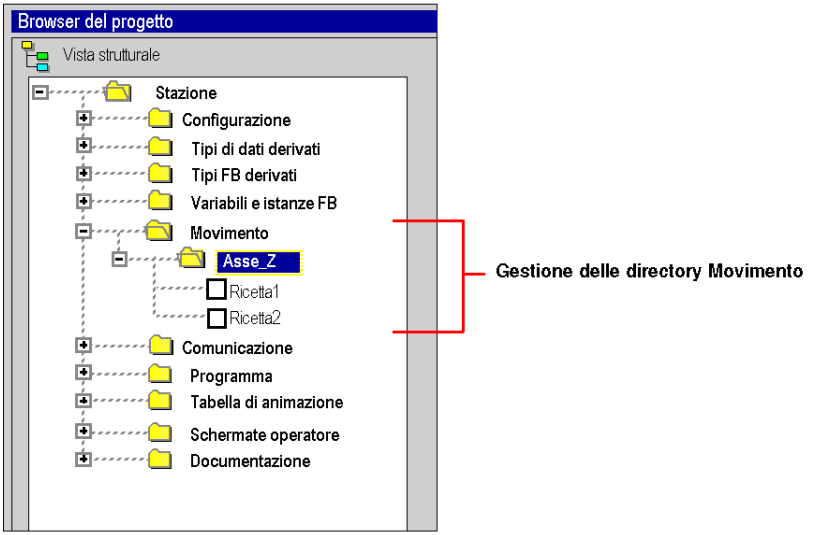
In breve

La directory **Motion** della vista strutturale del progetto permette di accedere alla dichiarazione e alla configurazione dei servoazionamenti.

Quando si dichiara un servoazionamento sono richieste varie informazioni, tra cui:

- il nome assegnato al servoazionamento
- il tipo di servoazionamento
- l'indirizzo CANopen del servoazionamento
- il codice di riferimento del servoazionamento
- la versione del servoazionamento
- l'immissione di nomi di variabili associati all'asse.

La figura seguente mostra un esempio di struttura della directory **Motion**:



In questo schema il nome assegnato al servoazionamento è 'Axis_Z'.

Per impostazione predefinita ogni volta che si crea un asse viene assegnata una ricetta. È possibile creare numerose ricette *(vedi pagina 68)*.

Comandi accessibili

La directory **Motion** consente di accedere ai seguenti servizi, raggiungibili tramite il menu contestuale:

Directory	Servizio
Motion	**Nuovo asse**: consente di creare un nuovo asse.
Asse	**Nuova ricetta**: consente di creare una nuova ricetta. **Elimina**: consente di eliminare un asse. **Proprietà**: consente di accedere alle proprietà dell'asse.
Ricetta	**Elimina**: consente di eliminare una ricetta. **Proprietà**: consente di accedere alle proprietà della ricetta.

Creazione e configurazione dell’asse

Informazioni generali

La directory **Motion** viene utilizzata per dichiarare un asse.

La procedura di creazione consente di semplificare la gestione e la programmazione di un asse tramite Unity Pro.

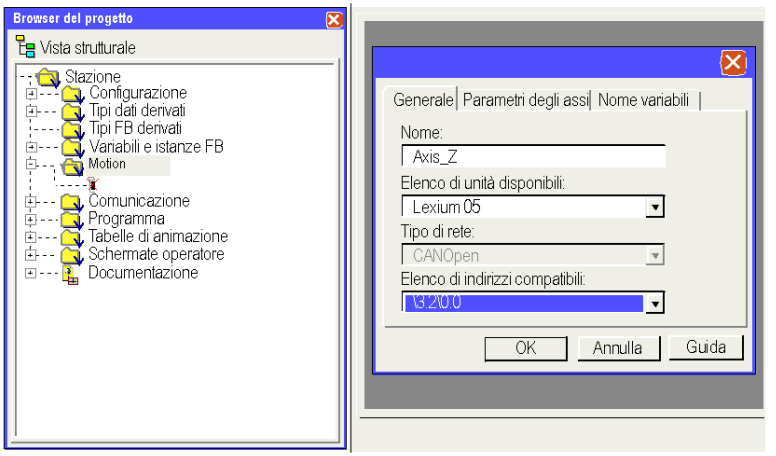
NOTA: Per qualsiasi modifica a un dispositivo sul bus CANopen, non è necessario riconfigurare i servoazionamenti non influenzati dalla modifica.

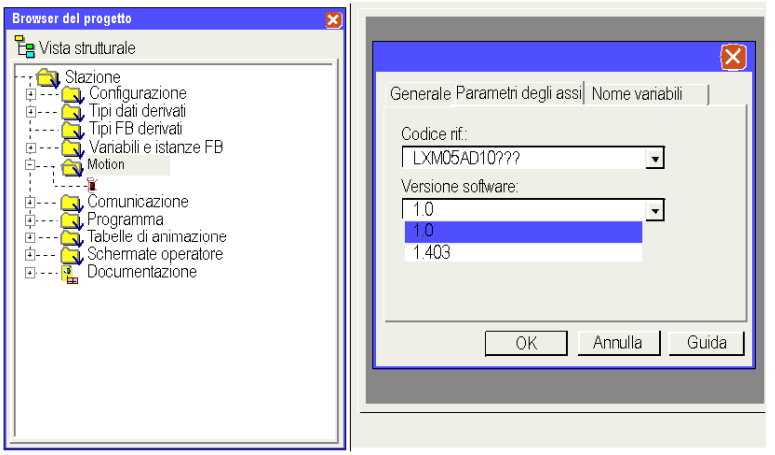
Creazione di un asse

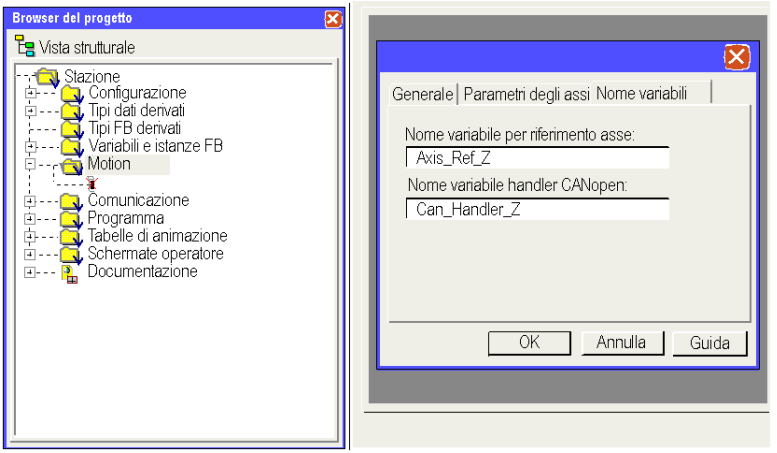
Eseguire queste azioni:

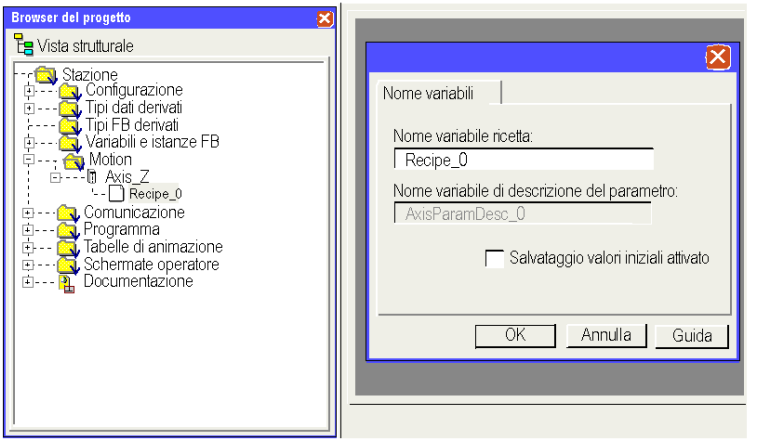
Passo	Azione
1	Fare clic con il pulsante destro sulla directory **Motion** ed eseguire il comando **Nuovo asse** dal menu contestuale.
2	Facendo clic sul comando **Nuovo asse**, si apre una finestra di dialogo con tre schede.

Passo	Azione

3	Nella scheda **Generale**,
	• immettere:
	  • un nome
	• selezionare:
	  • un servoazionamento dall'elenco
	  • un indirizzo compatibile CANopen
	
	Nota: se gli indirizzi CANopen non sono ancora stati definiti, lasciare **<Nessun collegamento>** nell'elenco. Se **<Nessun collegamento>** è assegnato a un indirizzo CANopen compatibile, è possibile proseguire con lo sviluppo dell'applicazione.
	Una volta definiti gli indirizzi CANopen, selezionare un indirizzo nell'elenco delle unità compatibili.
	In questa scheda, **Asse_Z** è configurato nel seguente modo:
	 

Passo	Azione
4	Nella scheda **Parametri degli assi**, selezionare: ● il codice di riferimento del servoazionamento ● la versione minima del firmware del servoazionamento In questa scheda, **Asse_Z** è configurato nel seguente modo:  **Nota:** si consiglia di verificare la coerenza tra la versione del firmware del servoazionamento e la versione dichiarata in Unity Pro. La versione consente di definire i parametri dell'azionamento. Durante l'inizializzazione del servoazionamento da parte del blocco funzione MFB `CAN_HANDLER`, la versione viene verificata.

Passo	Azione
5	Nella scheda **Nome variabili**, immettere: ● un nome per la variabile del tipo `Asse_Rif` associata al servoazionamento ● un nome per la variabile del tipo `Can_Handler` associata al servoazionamento In questa scheda, **Asse_Z** è configurato nel seguente modo: 
6	Fare clic su **OK** per confermare le selezioni.

Passo	Azione
7	Fare clic con il pulsante destro sulla sottodirectory **Ricetta_0** ed eseguire il comando **Proprietà** dal menu contestuale. A questo punto è possibile modificare le variabili della ricetta e dei parametri create per impostazione predefinita durante la creazione dell'asse. **Note:** se si contrassegna la casella di controllo Salvataggio valori iniziali - attivato la ricetta viene inclusa nell'applicazione. Questa funzionalità è disponibile per M340 V2.0 o versioni firmware successive, vedere la variabile della ricetta. *(vedi pagina 44)* In questa finestra le variabili per **Asse_Z** hanno i seguenti nomi predefiniti: 
8	Fare clic su **OK** per confermare la configurazione.

NOTA: È possibile creare più ricette per un asse (l'impostazione predefinita ne prevede una). Il caricamento della ricetta richiesta viene eseguito dal blocco TE_DOWNLOADDRIVEPARAMETER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*.

Il blocco della libreria MFB permette di:

- caricare i parametri in un nuovo servoazionamento se il precedente è difettoso;
- caricare una nuova ricetta in un servoazionamento durante una modifica della produzione, ad esempio

È possibile eliminare la ricetta se non si riesce ad usarla.

NOTA: Le dimensioni di memoria dei dati non identificati per la gestione di una ricetta in base al tipo di servoazionamento sono all'incirca di 2 Kword.

Variabili Axis_Ref, Can_Handler, AxisParamDesc e Recipe

In breve

Per ogni creazione di asse vengono create quattro variabili:

- Una variabile di tipo `Can_Handler` creata automaticamente dal browser motion, che può essere rinominata mediante la directory dell'asse
- Una variabile di tipo `Axis_Ref` che può essere rinominata mediante la directory dell'asse
- Una variabile di tipo tabella di byte (ARRAY[....] OF BYTE) che ha come nome predefinito `Recipe_x` (dove *x* è un valore) ma che può essere rinominata mediante la directory `Recipe_x`
- Una variabile di tipo tabella di interi senza segno (ARRAY[....] OF UINT) di nome `AxisParamDesc_x` (dove *x* è un valore), che non può essere rinominata

`Can_Handler`

Questa è una variabile di tipo EFB. Prende il nome in base alla variabile del gestore CANopen.

È dichiarata nella scheda **Nome variabile** durante la creazione dellßasse *(vedi pagina 39)*.

Deve essere usata nel programma come l'istanza del blocco funzione MFB `CAN_HANDLER` *(vedi pagina 59)*.

`Axis_Ref`

Questa è una variabile strutturata di tipo `AXIS_REF` che prende il nome in base alla variabile di riferimento dell'asse.

È dichiarata nella scheda **Nome variabile** durante la creazione dellßasse *(vedi pagina 39)*.

Deve essere specificata nel parametro di ingresso per ogni blocco MFB usato dall'asse.

AxisParamDesc

Questa è una variabile di tipo tabella di interi senza segno (ARRAY[....] OF UNIT). Viene creata automaticamente durante la creazione dell'asse *(vedi pagina 39)*. Prende il nome in base alla variabile di descrizione dei parametri che può essere vista nelle proprietà `Recipe_x` dell'asse.

Questa variabile deve essere specificata nel parametro di ingresso PARAMETERLIST *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* dei blocchi TE_UPLOADDRIVEPARAMETERR *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* e `TE_DOWNLOADDRIVEPARAMETER` della libreria MFB ed è utile per la creazione di ricette o per la sostituzione dell'asse se difettoso.

Questa variabile:
- non può essere modificata,
- è identica se gli assi dichiarati nell'applicazione hanno gli stessi riferimenti e la stessa versione del firmware.

Ricetta

Questa è una variabile di tipo tabella di byte (ARRAY[....] OF BYTE). Viene creata automaticamente durante la creazione dell'asse *(vedi pagina 39)*. Prende il nome in base alla variabile della ricetta che può essere vista nelle proprietà `Recipe_x` dell'asse.

Questa variabile deve essere specificata nel parametro di ingresso `PARAMETERSET` del blocco TE_UPLOADDRIVEPARAMETER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* o TE_DOWNLOADDRIVEPARAMETER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* della libreria MFB ed è utile per la creazione di ricette o per la sostituzione dell'asse se difettoso.

La variabile può essere modificata mediante le proprietà `Recipe_x` dell'asse.

La ricetta può essere inclusa nell'applicazione:

L'applicazione può essere aggiornata con una memorizzazione nei valori iniziali utilizzando il comando 'Aggiorna i valori iniziali con i valori correnti' o il bit %S94. Di conseguenza, il file STU o XEF includeranno i valori ottenuti dall'azionamento dopo una chiamata TE_Upload. Infine, contrassegnare la casella di controllo 'Salvataggio valori iniziali - attivato' per rendere disponibile questa funzionalità.

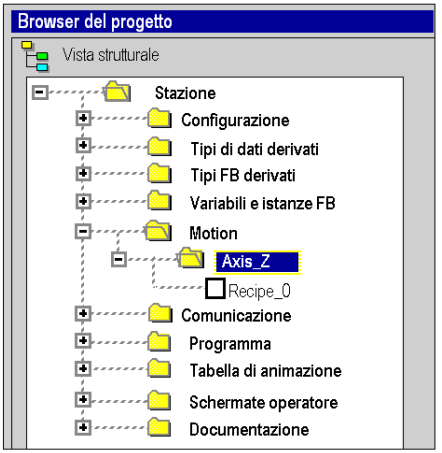
NOTA: Per impostazione predefinita, la casella di controllo Salvataggio valori iniziali - attivato non è contrassegnata.

NOTA: La funzionalità Salvataggio valori iniziali - attivato è disponibile per M340 V2.0 o versioni del firmware successive.

Risultato della configurazione della directory Motion

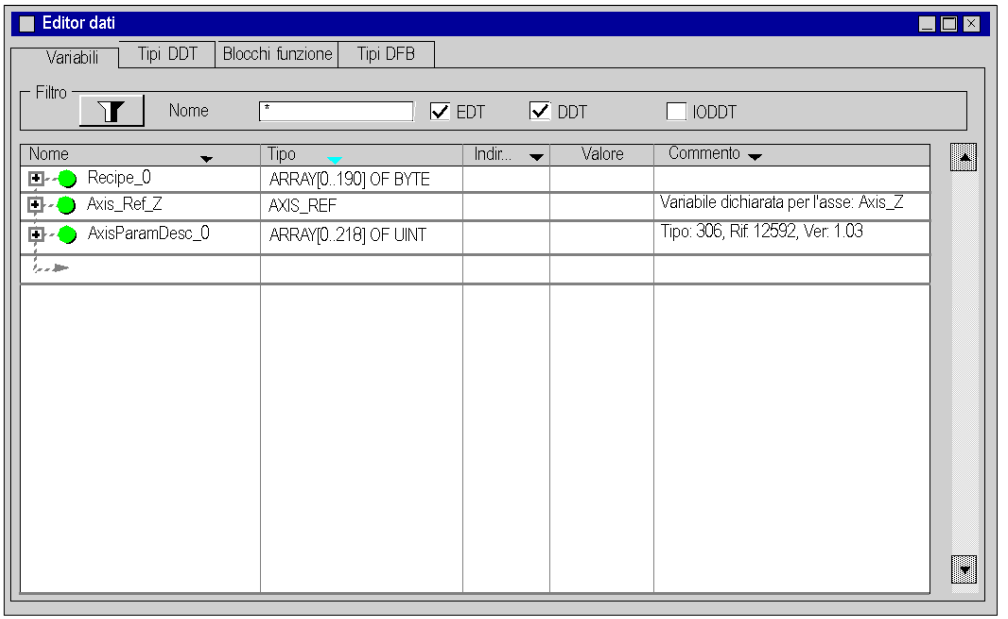
Nel browser del progetto

Nella seguente figura è illustrata la struttura ad albero della directory **Motion** dopo la configurazione:



Nell'editor dati

Nella seguente schermata sono indicate le variabili create nell'editor di dati durante la creazione degli assi. Per accedere a questa schermata, fare doppio clic nella directory **Variabili e istanze FB** nel browser del progetto:



Per accedere alla variabile `Can_Handler_Z`, fare clic sulla scheda **Blocchi funzione**.

2.5 Configurazione del Lexium 05

Scopo di questa sezione

Questa sezione descrive la configurazione di base del servoazionamento usando PowerSuite for **Lexium 05** e l'interfaccia utente del pannello frontale.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Configurazione di Lexium 05 in PowerSuite	49
Configurazione con l'interfaccia utente di Lexium 05	52

Configurazione di Lexium 05 in PowerSuite

Panoramica

Con PowerSuite, gli utenti possono definire le basi delle apparecchiature installate, descrivere le configurazioni associate e le impostazioni di comunicazione.

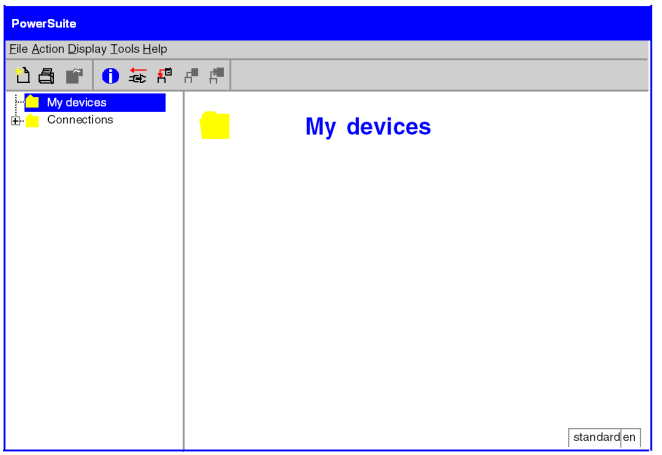
PowerSuite offre poi l'accesso a un gruppo di operazioni per la modifica o il trasferimento delle configurazioni nonché per la connessione alle apparecchiature.

Il principio di navigazione di PowerSuite permette di associare un'interfaccia di configurazione con ogni tipo di apparecchiatura rendendone possibile il controllo, la regolazione e il monitoraggio.

NOTA: I necessari segnali, es. LIMN, LIMP, REF devono essere collegati o disattivati dal software di regolazione.

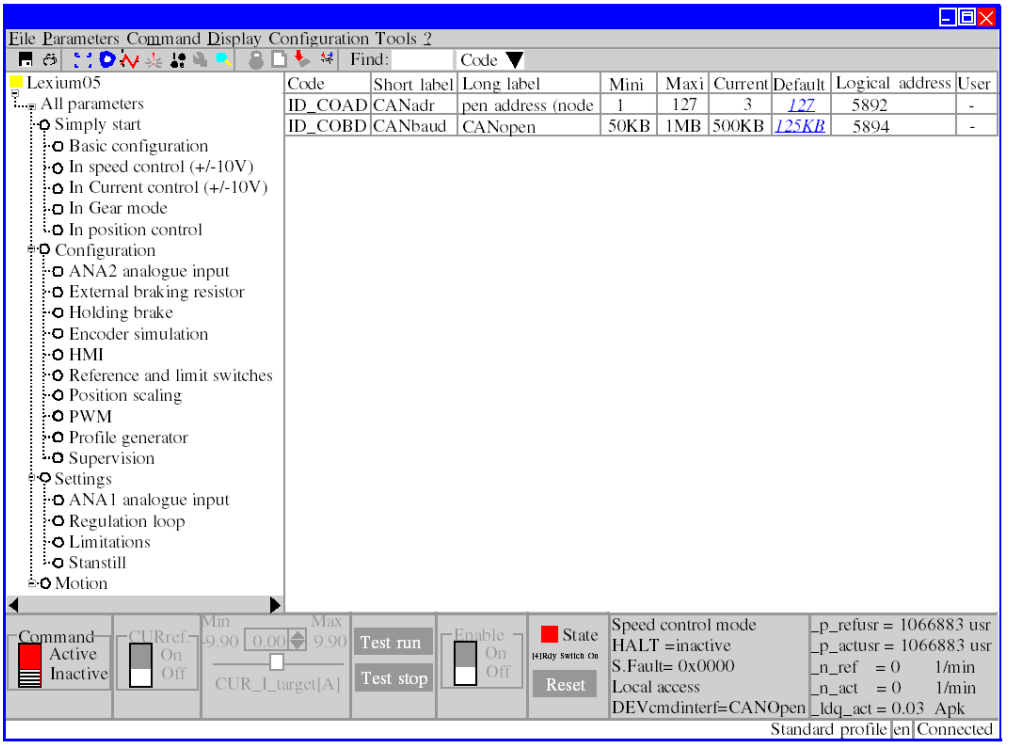
Connessione al Lexium 05

Questa tabella descrive la procedura di connessione al **Lexium 05**:

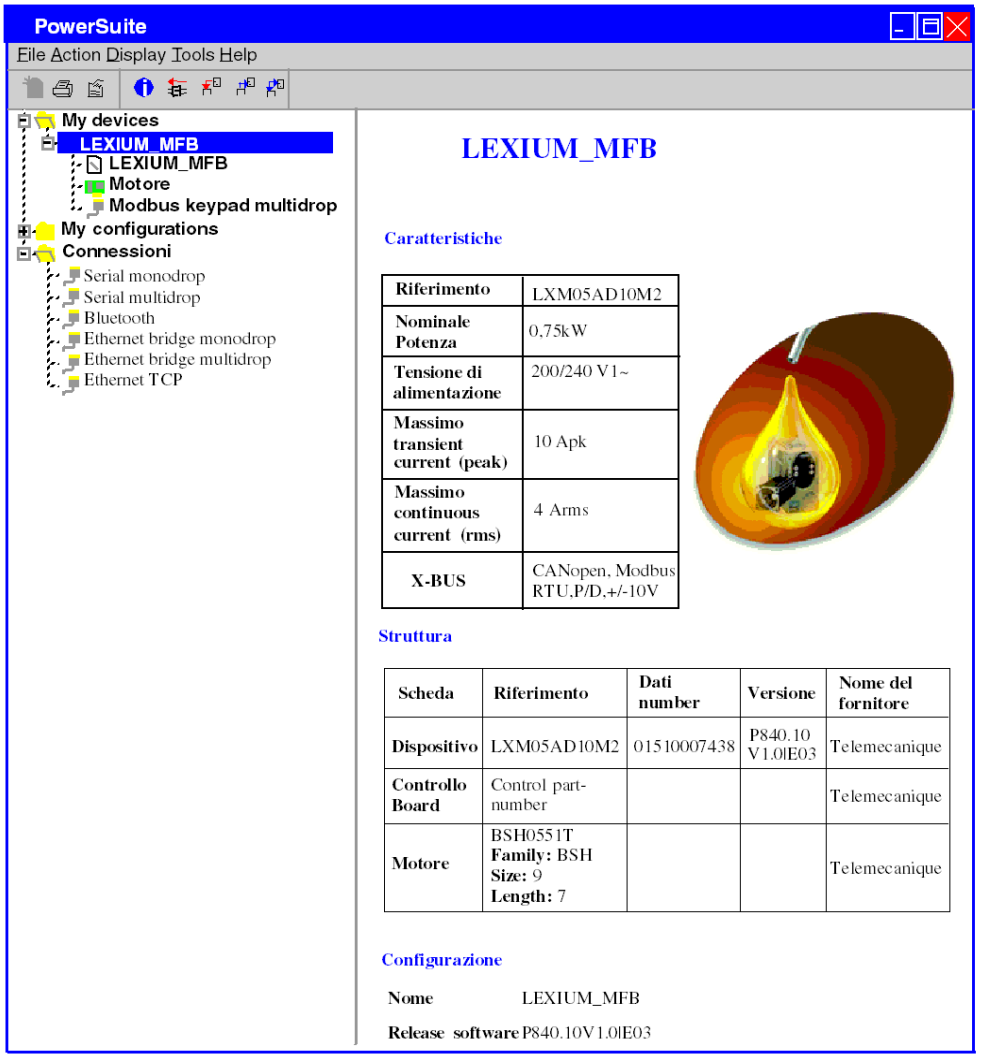
Passo	Azione
1	Collegare il PC, su cui è installato PowerSuite for **Lexium 05**al connettore **RJ45** del servoazionamento da configurare.
2	Avviare PowerSuite for **Lexium 05**, **Risultato:** viene visualizzata la schermata di avvio: 
3	Scegliere **Azione** quindi **Collegamento**. **Risultato:** viene visualizzata una casella di testo.
4	Digitare un nome di progetto (Lexium05_MFB) e fare clic su **OK**. **Risultato:** viene visualizzata una finestra di conferma del trasferimento.
5	Premere **Alt F** per avviare il trasferimento dati dal servoazionamento alla stazione di lavoro collegata.

Configurazione di base di Lexium 05

Questa tabella descrive la procedura d'immissione delle impostazioni di base:

Passo	Azione
1	Dopo la connessione e il trasferimento delle configurazioni delle apparecchiature, PowerSuite visualizza una schermata di configurazione in una nuova finestra che offre l'accesso al controllo dell'apparecchiatura e alle funzioni di regolazione e monitoraggio. Nella struttura ad albero visualizzata, scegliere **CANopen** nella directory *Communication*. **Risultato:** viene visualizzata la seguente finestra: 
2	Fare doppio clic sul valore nella riga **ID_COAD**, colonna **Valore corrente**, e digitare l'indirizzo CANopen di **Lexium 05**.
3	Fare doppio clic sul valore nella riga **ID_COBD**, colonna **Valore corrente** e scegliere la velocità in baud del bus CANopen.
4	Salvare le impostazioni CANopen nella EEprom con il comando **Configurazione** →**Salva nella EEprom**. **Nota:** è possibile regolare le impostazioni del servoazionamento con la stessa procedura.

Passo	Azione
5	Una volta regolate le impostazioni, usare il comando **Configurazione** →**Scollega** per scollegarsi. **Risultato:** viene visualizzata la seguente schermata, che mostra i dati salvati in locale:
6	Il Lexium 05 deve essere spento e poi riacceso in modo da rendere effettive le nuove impostazioni.



Configurazione con l'interfaccia utente di Lexium 05

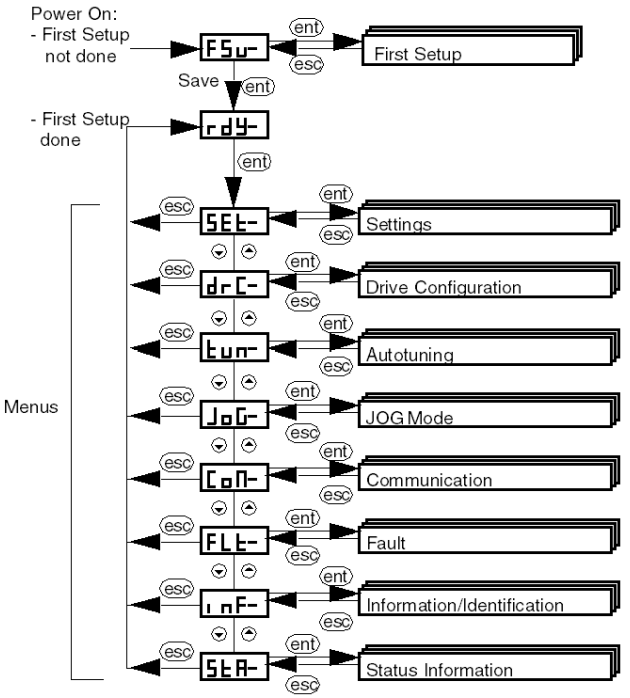
Panoramica

Nel **Lexium 05** è integrata un'interfaccia utente. Con questa interfaccia è possibile:

- mettere in linea l'apparecchiatura
- configurararla
- eseguire la diagnostica

Struttura del menu d'interfaccia

Il grafico seguente illustra la procedura d'accesso ai menu d'interfaccia utente:



Impostazioni di base

La seguente tabella descrive la procedura d'immissione delle impostazioni di base (indirizzo e velocità CANopen) con l'interfaccia.

Passo	Azione
1	Premere il pulsante **ENT** sull'interfaccia. **Risultato:** il menu **SET** (Setting) viene visualizzato sull'indicatore di stato dell'interfaccia.
2	Premere il pulsante più volte per accedere al menu **COM**. **Risultato:** il menu **COM** (Communication) viene visualizzato sull'indicatore di stato dell'interfaccia.
3	Premere il pulsante **ENT** sull'interfaccia. **Risultato:** il sottomenu **COAD** (CANopen Address) viene visualizzato sull'indicatore di stato dell'interfaccia.
4	Premere di nuovo **ENT**. **Risultato:** viene visualizzato un valore corrispondente all'indirizzo CANopen dell'apparecchiatura.
5	Premere il pulsante per diminuire il valore o il pulsante per aumentare il valore dell'indirizzo CANopen. Premere **ENT** quando l'indirizzo CANopen desiderato è visualizzato (3). **Risultato:** il valore viene confermato e viene visualizzato di nuovo il sottomenu **COAD** (indirizzo CANopen).
6	Premere una volta **ESC** per tornare al sottomenu **COAD**.
7	Premere il pulsante per accedere al sottomenu **COBD** (velocità in baud di CANopen). Premere **ENT**. **Risultato:** viene visualizzato un valore corrispondente alla velocità CANopen dell'apparecchiatura.
8	Premere il pulsante per diminuire il valore o il pulsante per aumentare il valore della velocità CANopen. Premere **ENT** quando viene visualizzata la velocità CANopen desiderata (500). **Risultato:** il valore viene confermato e viene visualizzato di nuovo il sottomenu **COBD** (velocità in baud CANopen).
9	Premere più volte **ESC** per ritornare al menu principale (**RDY** predefinito).

Programmazione applicazione

3

Argomento del capitolo

Questo capitolo descrive le varie fasi dello sviluppo del programma applicativo.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Dichiarazione delle variabili	56
Programmazione dell'esempio	57
Blocco funzione `CAN_HANDLER`	59
Gestione delle modalità di funzionamento e di arresto degli assi	62
Comando di movimento	63
Monitoraggio del movimento	65
Stato e codici di errore degli assi	66
Backup e trasferimento parametri del servoazionamento	68
Trasferimento del progetto tra il terminale e il PLC	69

Dichiarazione delle variabili

In breve

Oltre alle variabili associate all'asse quando questo viene creato nella directory **Motion**, è necessario dichiarare altre variabili.

Queste devono essere assegnate a:

- parametri di ingresso o di uscita dei blocchi MFB
- oggetti schermate operatore *(vedi pagina 77)*.

Le variabili consentono di utilizzare certi dati e di controllare l'asse con i blocchi della libreria MFB.

Dichiarazione nell'editor dei dati

La tabella seguente riassume le variabili che devono essere create nell'editor dei dati per l'esempio di esercitazione:

Nome	Tipo	Commento
`Cmd_Home_Z`	BOOL	Comando di ritorno dell'asse alla posizione di origine
`Cmd_Mvt_Z`	BOOL	Comando di movimento dell'asse
`Cmd_Run_Z`	BOOL	Comando di esecuzione dell'asse
`Cmd_Stop_Z`	BOOL	Comando di arresto dell'asse
`Cmd_Reset_Z`	BOOL	Comando di riconoscimento dell'asse
`Cmd_Upload_Z`	BOOL	Comando di salvataggio dei dati dell'asse in una tabella della ricetta
`Cmd_Download_Z`	BOOL	Comando di trasferimento dei dati da una tabella della ricetta all'asse
`Axis_OK_Z`	BOOL	Asse riconosciuto dal bus CANopen
`Position_Z`	DINT	Valore della posizione dell'asse
`Velocity_Z`	DINT	Valore della velocità dell'asse
`Recipe_Z`	ARRAY[0..190] OF BYTE	Variabile buffer per la gestione delle ricette
`CAN`	T_COM_CO_BMX	IODDT di gestione della porta CANOpen

NOTA: le dimensioni della tabella di gestione delle ricette sono conformi con quelle delle ricette create dalla directory **Motion**.

Programmazione dell'esempio

In breve

Dopo la dichiarazione e l'impostazione dei parametri dell'hardware, la seconda fase di sviluppo dell'esempio di esercitazione è rappresentata dalla programmazione del movimento.

La programmazione dell'asse si suddivide in:

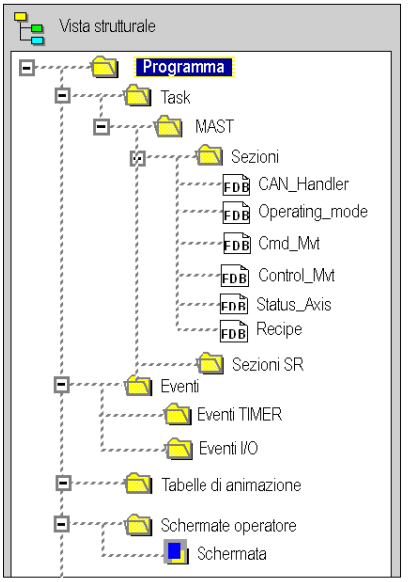
- dichiarazione delle variabili
- una schermata operatore che permette di visualizzare e controllare l'asse
- programmazione strutturata in più sezioni

Dichiarazione delle sezioni

La tabella seguente presenta un riepilogo delle sezioni di programma da creare.

Nome della sezione	Linguaggio	Descrizione
CAN_Handler *(vedi pagina 59)*	FBD	Questa sezione consente di verificare se i parametri dell'asse corrispondono alla realtà.
Operating_mode *(vedi pagina 62)*	FBD	Questa sezione consente di accendere i servoazionamenti e di verificare gli assi.
Cmd_Mvt *(vedi pagina 63)*	FBD	Questa sezione consente di impostare un punto di riferimento di origine per l'asse e quindi di controllarlo nel movimento assoluto.
Control_Mvt *(vedi pagina 65)*	FBD	Questa sezione consente di determinare la posizione e la velocità dell'asse.
Status_Axes *(vedi pagina 66)*	FBD	Questa sezione consente di determinare lo stato dell'asse e di eseguire la diagnostica per un evento.
Recipe *(vedi pagina 68)*	FBD	Questa sezione consente di salvare o ripristinare i dati di un servoazionamento.

Lo schema seguente mostra la struttura del programma dopo che sono state create le sezioni di programmazione:



Blocco funzione CAN_HANDLER

In breve

L'uso del blocco funzione **MFB** CAN_HANDLER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* è **essenziale** e **obbligatorio** nella programmazione dell'asse. La sezione di programma con questo blocco funzione **MFB** deve essere associata allo stesso task del master del bus CANopen *(vedi pagina 31)*.

Questo permette di verificare:

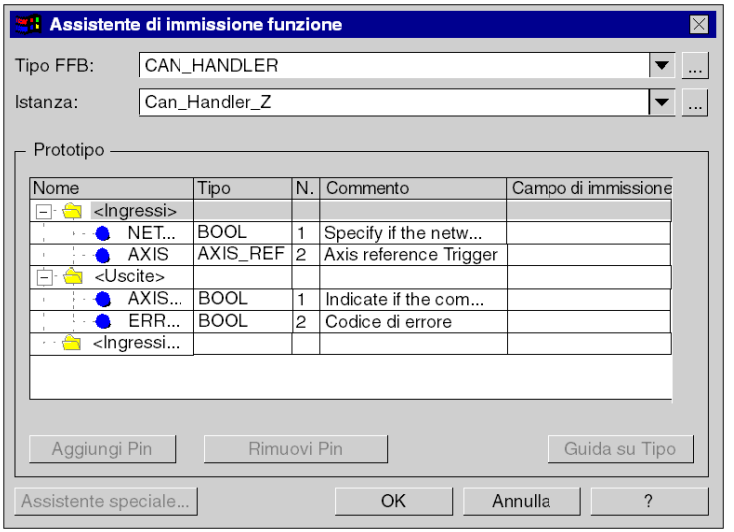
- la comunicazione CANopen
- la coerenza tra la configurazione del software e il dispositivo fisico collegato.

Questo blocco utilizza le due variabili che appartengono alla directory dell'asse. La variabile `Can_Handler_Z` deve essere usata come istanza e la variabile `Axis_Ref_Z` deve essere assegnata al parametro di ingresso del blocco `AXIS`.

Inserimento di un blocco e selezione dell'istanza

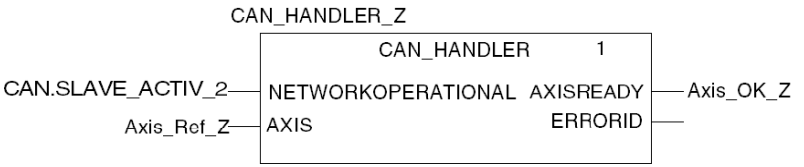
Questa tabella descrive la procedura per l'inserimento di un blocco e la selezione dell'istanza di un blocco in una sezione di programma:

Passo	Azione
1	Fare clic col pulsante destro in corrispondenza di un campo vuoto nella sezione FBD per visualizzare il menu contestuale.
2	Eseguire il comando **Assistente immissione FFB..** dal menu contestuale. **Risultato**: viene visualizzato l'Assistente di immissione funzione.
3	Fare clic sull'icona **...** nella riga **Tipo FFB**. **Risultato**: viene visualizzata la finestra **Selezione tipo FFB**.
4	Espandere **Librerie** →**MotionFunctionBlock** e fare clic su **MFB**. **Risultato**: tutti i blocchi della libreria **MotionFunctionBlock** vengono visualizzati sul lato destro della finestra **Selezione tipo FFB**.
5	Selezionare il blocco `CAN_HANDLER` e confermare la scelta facendo clic su **OK**. **Risultato**: Viene visualizzata la finestra **Assistente immissione FFB..**, configurata dal blocco `CAN_HANDLER`.
6	Fare clic sull'icona **...** nella riga **Istanza**. **Risultato**: viene visualizzata la finestra **Selezione istanza FB**.

Passo	Azione
7	Selezionare l'istanza `Can_Handler_Z` e confermare la scelta facendo clic su **OK**. **Risultato**: la variabile `Can_Handler_Z` viene visualizzata nel campo **Istanza**: 
8	Confermare la configurazione del blocco facendo clic su **OK**. **Risultato**: la sezione FBD viene nuovamente visualizzata. Un simbolo viene aggiunto al cursore del mouse.
9	Fare clic su un campo vuoto nella sezione FBD. **Risultato**: il blocco `CAN_HANDLER`, istanziato dalla variabile `Can_Handler_Z`, viene inserito nella sezione FBD.
10	Specificare i parametri di ingresso e di uscita come definiti nel sommario.

Contenuto

La schermata seguente mostra il risultato della sezione:



CAN.SLAVE_ACTIV_2 corrisponde al bit slave attivo creato da IODDT T_COM_CO_BMX.

Il parametro di ingresso NETWORKOPERATIONAL deve essere assegnato a un bit che convalida il funzionamento corretto della rete CANopen.

L'assegnazione di questo parametro è lasciata alla discrezione dello sviluppatore. Dipende dalla filosofia del processo e dal modo in cui viene gestito il bus.

Ad esempio, questo parametro può essere collegato a un oggetto o a un'equazione IODDT di tipo T_COM_CO_BMX *(vedi Modicon M340 con Unity Pro, CANopen, Manuale dell'utente)*.

Gestione delle modalità di funzionamento e di arresto degli assi

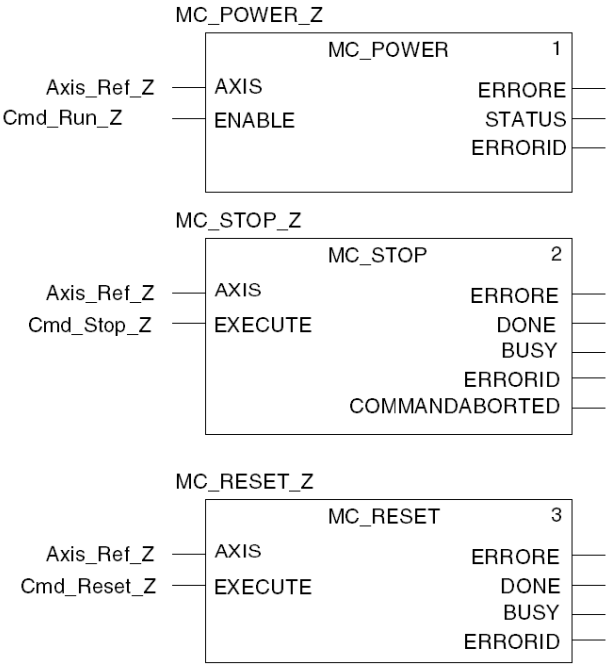
In breve

Questa sezione comprende i seguenti blocchi MFB:

- MC_POWER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*, che permette di attivare o disattivare i servoazionamenti
- MC_STOP *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*, che permette di arrestare qualsiasi movimento in corso
- MC_RESET *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*, che permette di inizializzare i blocchi funzione e di riconoscere i guasti del servoazionamento.

Sommario

La schermata mostra una parte della sezione da sviluppare:



I blocchi vengono istanziati all'ingresso variabili direttamente nella zona **Istanza** dell'**Assistente immissione FFB** per facilitare la diagnostica successiva utilizzando tabelle di animazione.

Comando di movimento

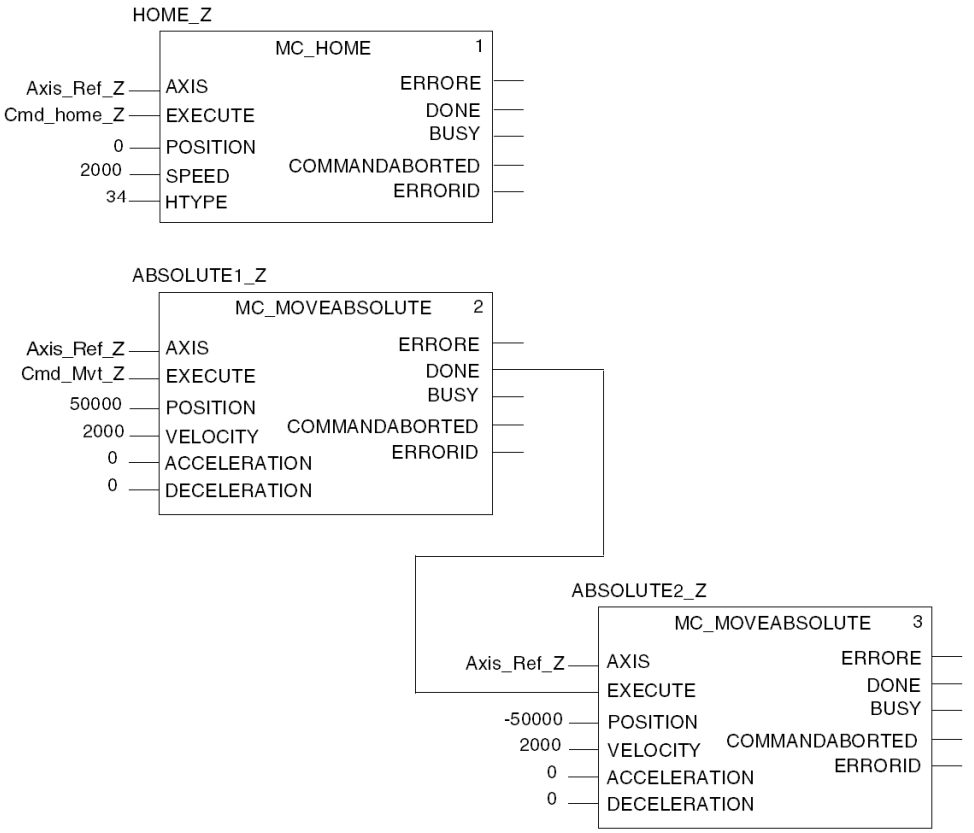
In breve

Questa sezione di programmazione si riferisce ai seguenti blocchi MFB:

- MC_HOME *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*, che permette di impostare un punto di riferimento di origine per l'asse prima di avviare l'asse in movimento assoluto
- MC_MOVEABSOLUTE *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*, che permette all'asse di eseguire un movimento assoluto.

Sommario

La schermata mostra questa parte della sezione:



Per l'esempio di esercitazione, la sezione è costituita da un tipo di sequenza di movimenti reversibili.

Il movimento verso l'esterno è determinato dal bit `Cmd_Mvt_Z` della schermata operatore *(vedi pagina 77)*.

Il movimento di ritorno è determinato dalla fine del movimento verso l'esterno.

L'unità di posizione è **USR** e l'unità di velocità è **rpm**.

Il valore del tipo di ricerca del punto di riferimento `HTYPE` (34) corrisponde a un movimento verso il punto di riferimento di un solo giro, in direzione di rotazione positiva.

Monitoraggio del movimento

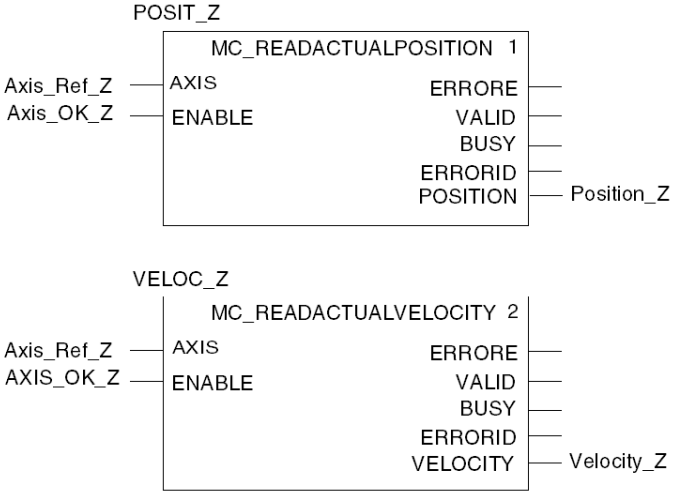
In breve

Questa sezione tratta dei blocchi MFB MC_READACTUALPOSITION *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* e MC_READACTUALVELOCITY *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*.

Questi blocchi sono utilizzati per determinare la posizione esatta e la velocità dell'asse.

Sommario

La schermata mostra una parte della sezione da sviluppare:



Quando il bit `Axis_OK_Z` è attivato, i valori di posizione e di velocità sono continuamente visualizzati sulla schermata operatore *(vedi pagina 77)*.

Stato e codici di errore degli assi

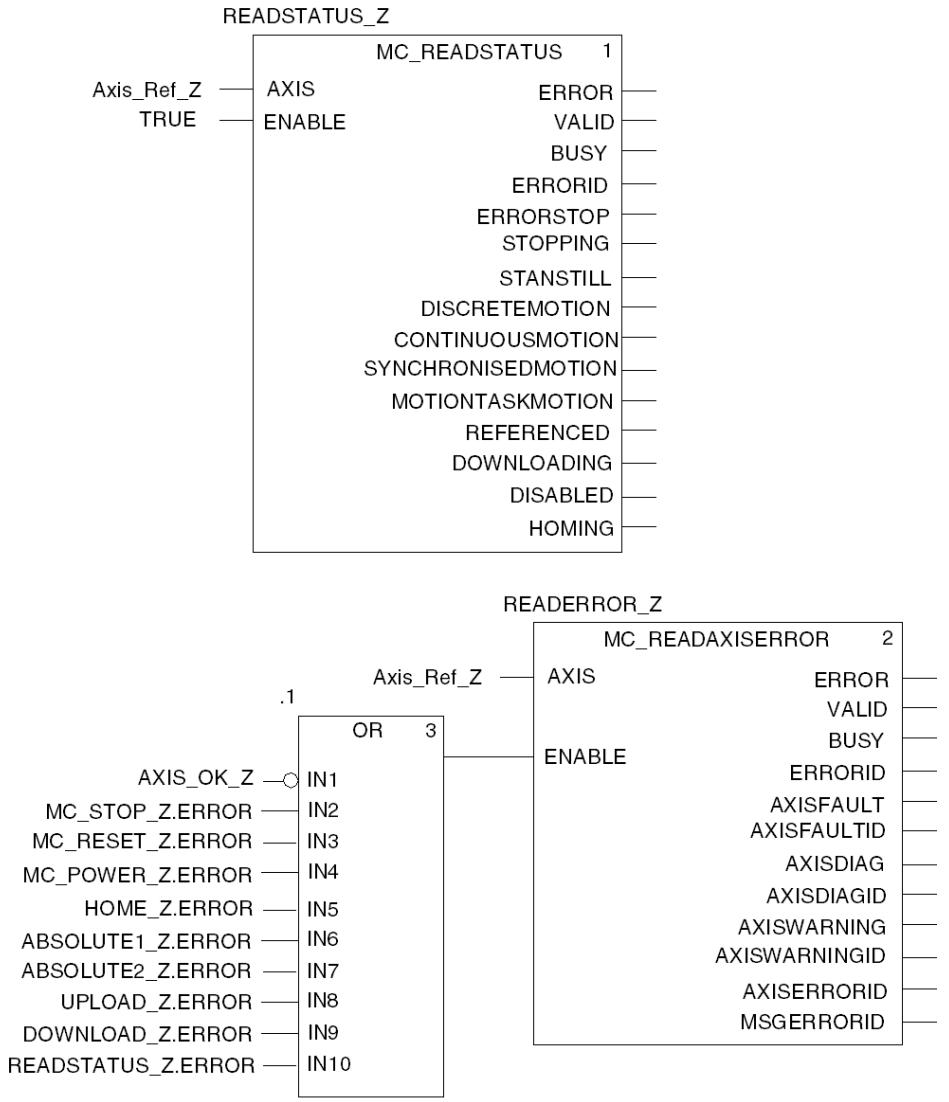
In breve

Questa sezione comprende i seguenti blocchi MFB:

- MC_READSTATUS *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*, che permette di determinare lo stato dell'azionamento *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*
- MC_READAXISERROR *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*, che permette di determinare i valori degli errori in base al tipo di errore verificatosi nell'azionamento e di individuarne le cause *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*.

Sommario

La schermata mostra una parte della sezione:



Le variabili `UPLOAD_Z.ERROR` e `DOWNLOAD_Z.ERROR` devono essere aggiunte al blocco `OR` dopo aver creato la sezione ricetta *(vedi pagina 68)*.

Backup e trasferimento parametri del servoazionamento

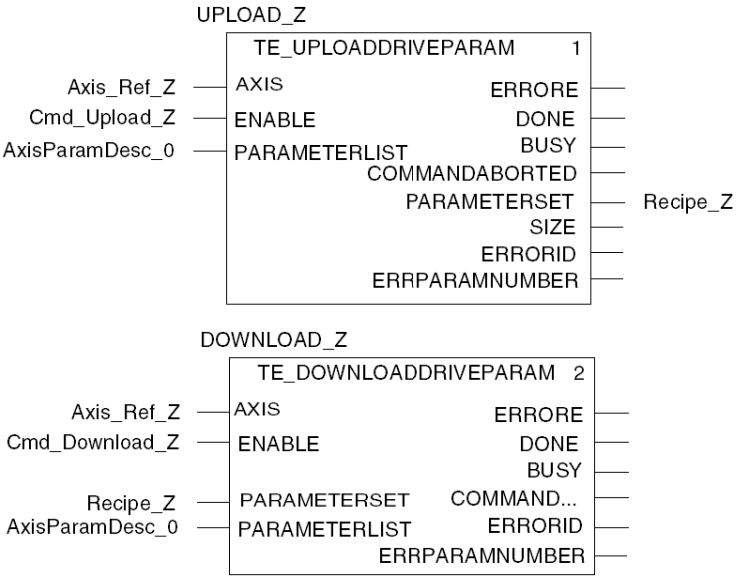
In breve

Questa sezione di programmazione tratta dei seguenti blocchi MFB:

- TE_UPLOADDRIVEPARAM *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*, utilizzato per eseguire il backup della configurazione di un servoazionamento in una tabella dati.
- TE_DOWNLOADDRIVEPARAM *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*, utilizzato per trasferire i parametri della tabella dati a un servoazionamento.

Sommario

La schermata qui sotto mostra la sezione Ricetta:



Se è attivato `Cmd_Upload_Z`, la configurazione del servoazionamento viene salvata nella tabella dati `Recipe_Z` (variabile del buffer per le ricette).

Se è attivato `Cmd_Download_Z`, la configurazione del servoazionamento viene ripristinata dalla tabella dati `Recipe_Z`.

Trasferimento del progetto tra il terminale e il PLC

In breve

Il trasferimento di un progetto consente di copiare il progetto corrente dal terminale alla memoria del PLC corrente (ovvero quello il cui indirizzo è selezionato).

Analisi e generazione di un progetto

Per eseguire contemporaneamente l'analisi e la generazione di un progetto, procedere come segue:

Passo	Azione
1	Attivare il comando **Ricrea tutto il progetto** del menu **Crea**. **Risultato:** il progetto viene analizzato e generato dal software.
2	Eventuali errori rilevati vengono visualizzati nella finestra delle informazioni situata al fondo dello schermo.

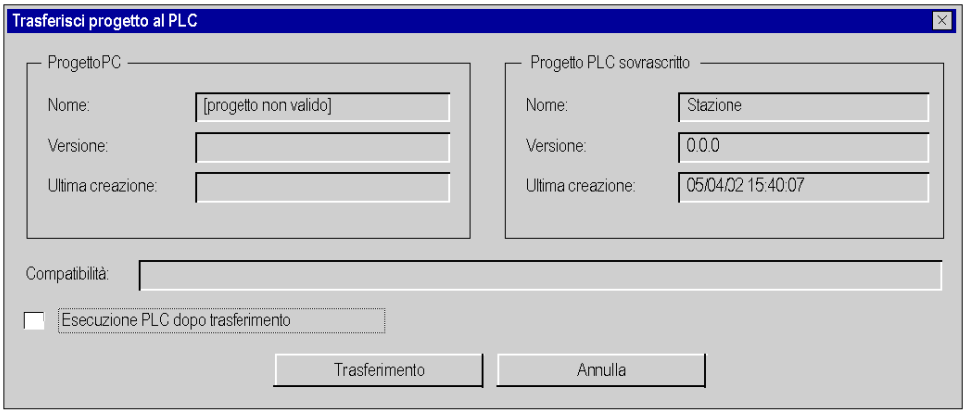
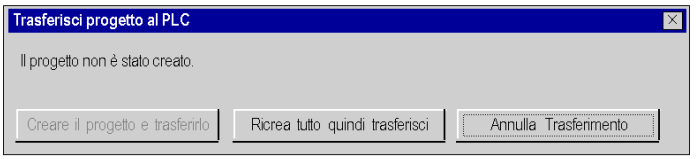
Backup del progetto

Per eseguire un backup del progetto, procedere come segue:

Passo	Azione
1	Selezionare il comando **Salva con nome** nel menu **File**.
2	Se necessario, selezionare la directory in cui verrà salvato il progetto (unità disco e percorso).
3	Immettere il nome del file: **MFB_Lexium05**.
4	Confermare con **Salva**. **Risultato:** il progetto viene salvato come **MFB_Lexium05.STU**.

Trasferimento del progetto al PLC

Per trasferire il progetto corrente al PLC, procedere come segue:

Passo	Azione
1	Utilizzare il comando **PLC →Definisci indirizzo**. Immettere **SYS** se si sta usando un supporto **USB** collegato direttamente tra PC (terminale) e PLC.
2	Passare alla modalità online con il comando **PLC →Connessione**.
3	Attivare il comando **PLC →Trasferisci progetto a PLC**. **Risultato**: viene visualizzata la schermata utilizzata per trasferire il progetto dal terminale al PLC: 
4	Selezionare il comando **Trasferimento**.
5	Se il progetto non è stato generato in anticipo, verrà visualizzata la seguente schermata, che permette di generarlo prima del trasferimento (**Ricrea tutto quindi trasferisci**) o di interrompere il trasferimento (**Annulla trasferimento**). 
6	L'avanzamento dell'operazione di trasferimento viene visualizzato sullo schermo. In qualsiasi momento, è possibile interrompere il trasferimento premendo il tasto **Esc**. In questo caso, il progetto PLC non sarà confermato. **Nota**: il trasferimento del progetto in una scheda di memoria Flash Eprom può richiedere parecchi minuti.

Debug dell’applicazione

4

Argomento del capitolo

Questo capitolo descrive le possibilità di debug dell’applicazione medante Unity Pro e PowerSuite for **Lexium 05**.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Regolazione del Lexium 05 con PowerSuite	72
Uso dei dati con le tabelle di animazione	73
Debug del programma	75
Uso dei dati con le schermate operatore	77

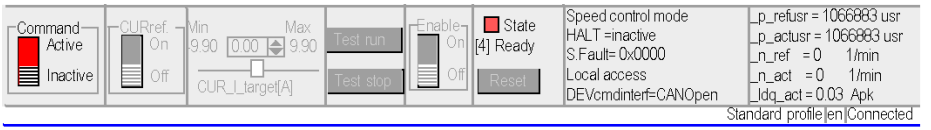
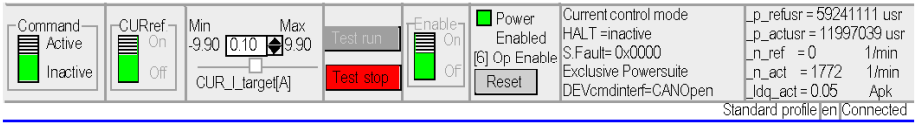
Regolazione del Lexium 05 con PowerSuite

Preliminarmente

Si raccomanda di regolare la cinematica dell'asse prima che il programma lo avvii automaticamente.

Esempio di regolazione

La seguente tabella offre un esempio di regolazione della cinematica:

Passo	Azione
1	Collegarsi *(vedi pagina 49)* al **Lexium 05**.
2	Dopo la connessione e il trasferimento delle configurazioni delle apparecchiature, PowerSuite apre una nuova finestra con la schermata di configurazione che offre l'accesso al controllo dell'apparecchiatura e alle funzioni di regolazione e monitoraggio. La seguente figura mostra parte della nuova finestra. Questa finestra più in basso permette d'accedere alle funzioni di comando del **Lexium 05**: 
3	Posizionare il cursore della zona **Comando** su **Attivo**.
4	Posizionare il cursore della zona **Enable** su **On**.
5	Fare clic sul pulsante **Reset** per azzerare tutti i problemi.
6	Fare clic sul pulsante **Test Run**.
7	Immettere il valore 0,1 nella zona **CUR_I_target**.
8	Posizionare il cursore della zona **CURref** su **On**. **Risultato:** il motore funziona e la sottofinestra è animata: 
9	Posizionare il cursore della zona **Comando** su **Inattivo** una volta che la regolazione è terminata.

Uso dei dati con le tabelle di animazione

In breve

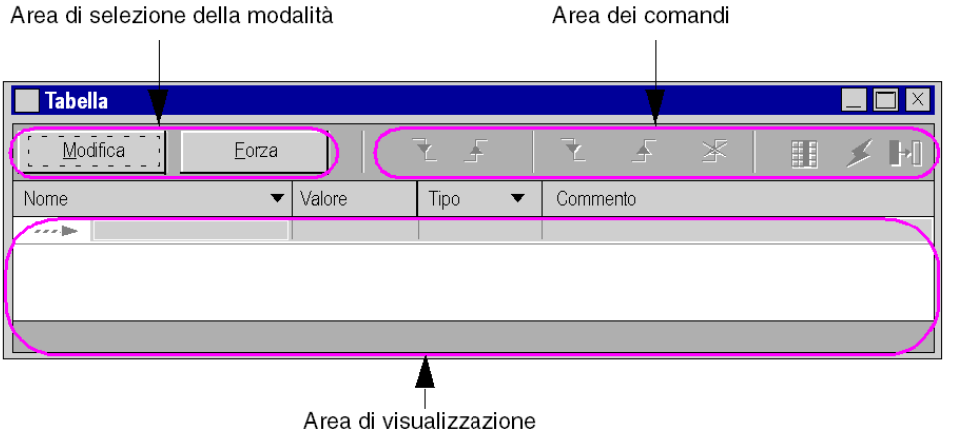
La tabella di animazione è lo strumento di base Unity Pro che permette di visualizzare e forzare lo stato delle variabili.

NOTA: Unity Pro offre anche uno strumento grafico chiamato **Schermate operatore** creato per facilitare l'uso dell'applicazione *(vedi pagina 77)*.

Le tabelle di animazione sono divise in 3 aree che includono:

- area **Modalità**
- area **Comando**
- area **Visualizzazione**

Tabella di animazione:



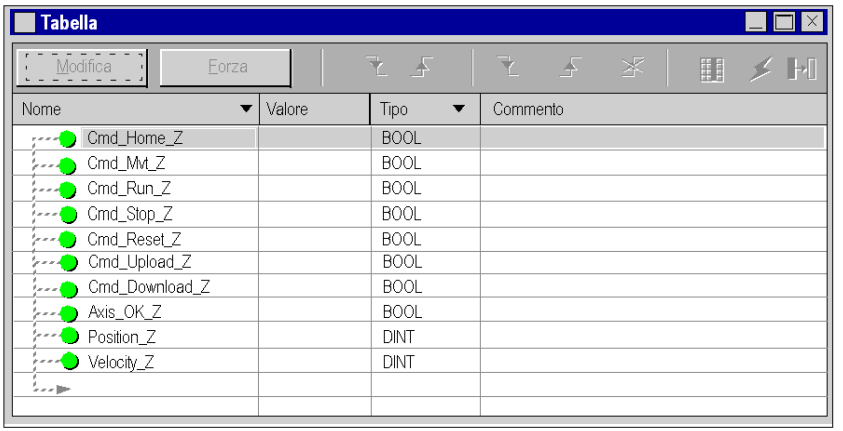
Creazione di una tabella di animazione

La tabella qui sotto presenta la procedura per la creazione di una tabella di animazione:

Passo	Azione
1	Fare clic con il pulsante destro del mouse sulla directory **Tabelle di animazione** nel browser progetto. **Risultato**: viene visualizzato il menu contestuale.
2	Selezionare **Nuova tabella di animazione**. **Risultato**: viene visualizzata una finestra delle proprietà della tabella.
3	Fare clic su OK per creare la tabella, a cui viene un nome predefinito. **Risultato**: viene visualizzata la tabella di animazione.

Aggiunta di dati alla tabella di animazione

La tabella di sotto presenta la procedura di creazione dati per visualizzare o forzare la tabella di animazione:

Passo	Azione
1	Nella finestra **Tabella**, fare clic sulla riga vuota nella colonna **Nome**.
2	Esistono due modi possibili per aggiungere dati: ● Immettere direttamente la variabile ● Fare clic sull'icona [...] per visualizzare la finestra di selezione dell'istanza in modo da poter visualizzare la variabile
3	Immettere o selezionare le variabili rispettive. ● `Cmd_Home_Z` per emettere un comando di ritorno dell'asse alla posizione di origine (home) ● `Cmd_Mvt_Z` per emettere un comando di movimento dell'asse ● `Cmd_Run_Z` per emettere un comando di esecuzione dell'asse ● `Cmd_Stop_Z` per emettere un comando di arresto dell'asse ● `Cmd_Reset_Z` per emettere un comando di riconoscimento dell'asse ● `Cmd_Upload_Z` per emettere un comando di salvataggio dei dati dell'asse in un tabella della ricetta ● `Cmd_Download_Z` per emettere un comando di trasferimento dei dati da una tabella della ricetta all'asse ● `Axis_OK_Z` per visualizzare l'asse riconosciuto dal bus CANopen ● `Position_Z` per determinare il valore della posizione dell'asse ● `Velocity_Z` per determinare il valore della velocità dell'asse **Risultato**: la tabella di animazione è simile alla seguente. 

Debug del programma

In breve

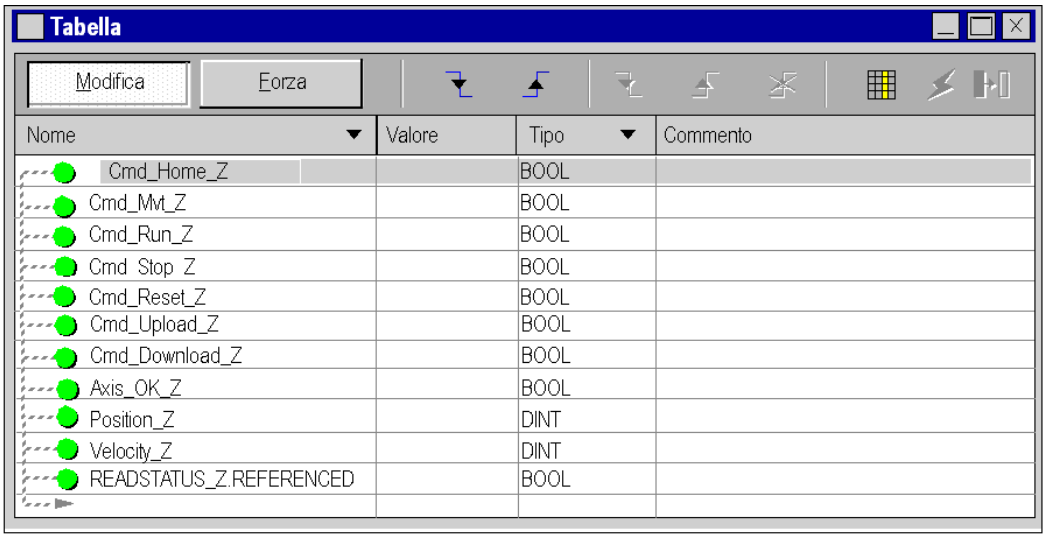
Dopo il trasferimento del programma e l'esecuzione dell'asse mediante Powersuite for **Lexium 05**, il processo viene messo in servizio.

La tabella di animazione è una soluzione di messa in servizio che consente di monitorare, modificare e/o forzare i valori delle variabili.

È possibile accedere ai set di parametri dell'asse e modificarli in Unity Pro mediante i blocchi di messaggeria MFB MC_READPARAMETER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* e MC_WRITEPARAMETER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*.

Modalità modifica

La schermata seguente mostra la tabella di animazione in modalità modifica:



Questa tabella permette di determinare lo stato dei parametri di ingresso e di uscita del blocco `MC_POWER`.

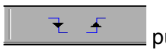
Per accedere a questa modalità, fare clic sul pulsante **Modifica** nella zona di selezione della modalità.

NOTA: questa operazione può essere assegnata ad altri blocchi funzione.

NOTA: la tabella di animazione è dinamica soltanto in modalità online (visualizzazione dei valori delle variabili).

Modifica di valori

Nell'esempio di esercitazione vengono utilizzate variabili booleane. Per modificare un valore booleano, procedere come segue:

Passo	Azione
1	Usare il mouse per selezionare la variabile booleana che si desidera modificare.
2	Fare clic sul  pulsante corrispondente al valore desiderato oppure eseguire i comandi **Imposta su 0** o **Imposta su 1** dal menu contestuale.

Avvio del sistema

La tabella seguente descrive la procedura di avvio del sistema usata nell'esempio:

Passo	Azione
1	Impostare la variabile `Cmd_Run_Z` su 1. **Risultato:** la variabile `Axis_OK_Z` passa a 1.
2	Impostare la variabile `Cmd_Reset_Z` su 1.
3	Impostare la variabile `Cmd_Home_Z` su 1. **Risultato:** l'asse è con riferimento.
4	Per ruotare l'asse, impostare la variabile `Cmd_Mvt_Z` su 1. **Risultato:** l'asse inizia a ruotare e i valori delle variabili `Position_Z` e `Velocity_Z` non sono più impostati su 0.
5	Per arrestare la rotazione dell'asse: • impostare la variabile `Cmd_Stop_Z` su 1. • impostare la variabile `Cmd_Mvt_Z` su 0. **Risultato :** l'asse arresta la rotazione.
6	Per riavviare la rotazione dell'asse e completare il movimento: • impostare la variabile `Cmd_Stop_Z` su 0. • impostare la variabile `Cmd_Mvt_Z` su 1. **Risultato:** l'asse inizia nuovamente a ruotare e completa il movimento.

Uso dei dati con le schermate operatore

In breve

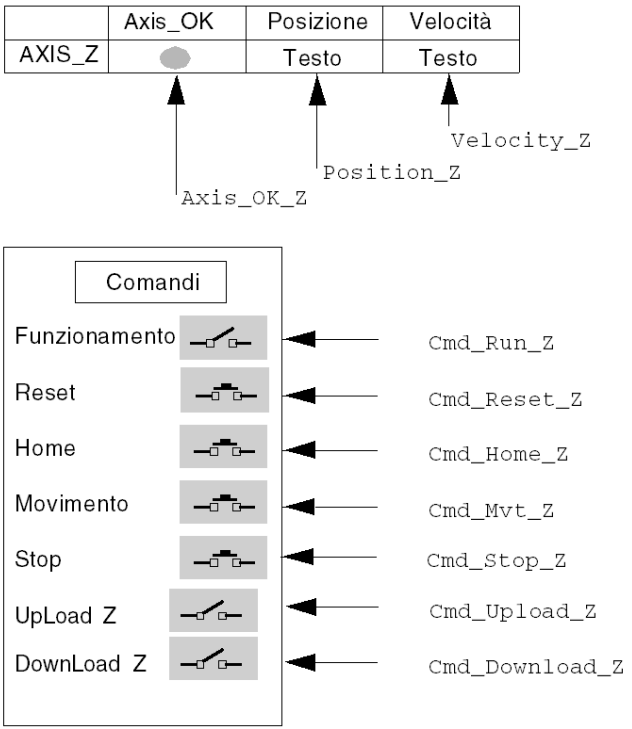
Quando si crea un progetto con schede d'ingresso, schede d'uscita o supervisione, la schermata operatore di Unity Pro (associata a parole e bit non allocati) consente di eseguire il il debug iniziale del programma.

Nell'esercitazione d'esempio, la schermata operatore è utilizzata per:

- visualizzare i dati dai servoazionamenti
- inviare comandi ai servoazionamenti

Rappresentazione

Nella seguente figura è schematizzato l'esempio operativo utilizzato per controllare l'asse e indicare le variabili da assegnare agli oggetti (pulsante, LED e testo):



Uso dell'applicazione

5

Gestione delle ricette

In breve

I blocchi TE_UPLOADDRIVEPARAM *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* e TE_DOWNLOADDRIVEPARAM *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* consentono di gestire le ricette di produzione.

Questa sezione contiene un esempio di procedura per la creazione e la gestione di ricette.

NOTA: per le macchine flessibili è possibile gestire più ricette di parametri.

Creazione e backup di ricette

La tabella descrive la procedura per la creazione di ricette:

Passo	Azione
1	Creare le ricette *(vedi pagina 39)* utilizzando la directory **Axis_Z**. **Risultato:** nuove variabili di ricette (`Recipe_0`, `Recipe_1`, ecc.) vengono create automaticamente nell'Editor dati *(vedi pagina 47)*.
2	Creare una variabile corrispondente al tipo di variabili di ricette. Questa variabile si chiama `Recipe_Z` nell'esempio di esercitazione. `Recipe_Z` funge da buffer quando si esegue il backup o il trasferimento di dati. **Nota**: è essenziale selezionare **Consenti array dinamici [ANY_ARRAY_XXX]** che si trova in **Strumenti →Opzioni del progetto →: Estensioni del linguaggio →Zona: Tipo di dati** per poter utilizzare le variabili di tipo tabella come le ricette.
3	Configurare i parametri del servoazionamento mediante Powersuite *(vedi pagina 49)*. Queste impostazioni iniziali permettono di configurare una ricetta.

Passo	Azione
4	Eseguire un backup dei parametri utilizzando il blocco TE_UPLOADDRIVEPARAM *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* nella variabile del buffer `Recipe_Z`. Il backup è riuscito correttamente se i bit del blocco MC_READSTATUS *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* sono come segue: ● DOWNLOADING *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* è impostato su 0 ● STANDSTILL *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* è impostato su 1
5	Trasferire i dati del backup dalla variabile del buffer `Recipe_Z` alla variabile `Recipe_0`.

6	Ripetere i passi 3 e 4 per trasferire i dati del backup dalla variabile del buffer `Recipe_Z` alla variabile `Recipe_1`.
	La programmazione seguente riporta un esempio di trasferimento di dati basato sul valore di `PRODUCTION`:
	`IF UPLOAD_Z.DONE AND PRODUCTION=0 THEN`
	`Recipe_0:=Recipe_Z;`
	`END_IF;`
	`IF UPLOAD_Z.DONE AND PRODUCTION=1 THEN`
	`Recipe_1:=Recipe_Z;`
	`END_IF;`

Trasferimento di dati dalle ricette

La tabella descrive la procedura per il trasferimento di dati delle ricette al servoazionamento (ad es. per una modifica di produzione):

Passo	Azione

1	Ricaricare la variabile del buffer `Recipe_Z` in base al valore di `PRODUCTION` (tipo di produzione richiesto).
	`IF Cmd_Download_Z AND PRODUCTION=0 THEN`
	`Recipe_Z:=Recipe_0;`
	`END_IF;`
	`IF Cmd_Download_Z AND PRODUCTION=1 THEN`
	`Recipe_Z:=Recipe_1;`
	`END_IF;`

2	Trasferire i dati dei parametri al servoazionamento utilizzando il blocco `TE_DOWNLOADDRIVEPARAM`della variabile del buffer TE_DOWNLOADDRIVEPARAM *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*.
3	Il trasferimento è riuscito correttamente se i bit del blocco MC_READSTATUS *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* sono come segue: ● DOWNLOADING *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* è impostato su 0 ● STANDSTILL *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)* è impostato su 1

Manutenzione dell’applicazione

6

Argomento del capitolo

Questo capitolo descrive la procedura da adottare per la sostituzione di un servoazionamento dopo aver diagnosticato un guasto.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sottosezioni:

Argomento	Pagina
Esempio di errore	82
Sostituzione di un servoazionamento guasto	84

Esempio di errore

In breve

La funzione `MC_ReadAxisError` consente di recuperare gli errori del sistema.

Se si verifica un errore o un'avvertenza, il blocco specifica un codice applicando un valore ai parametri di uscita `AXISFAULTID`, `AXISDIAGID` e `AXISWARNINGID`.

Codici di errore

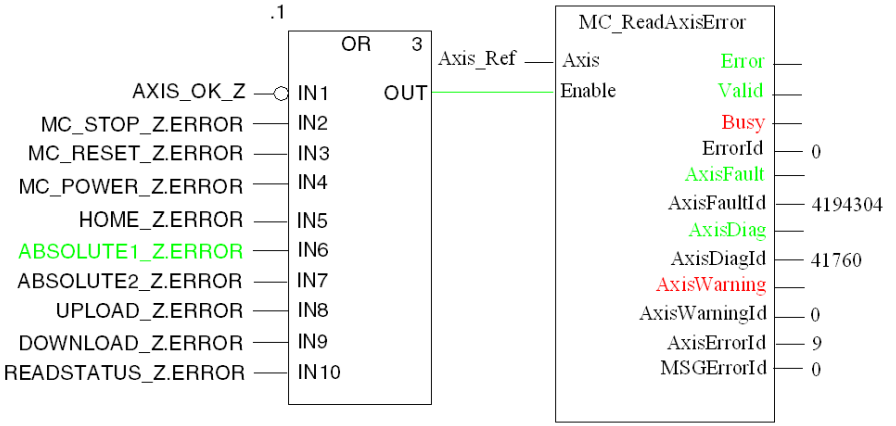
La tabella seguente mostra i codici di errore del **Lexium 05**:

	Lexium 05
`AxisFaultId`	SigLatched 301C:08
`AxisDiagId`	WarnLatched 301C:0C
`AxisWarningId`	StopFault 603F:0

NOTA: per identificare l'errore, consultare la documentazione CANopen per **Lexium 05**.

Ricerca degli errori

La tabella seguente descrive la procedura di ricerca degli errori in seguito a un codice di errore o a un avviso.

Passo	Azione
1	Il parametro di uscita AxisFault è impostato a 1. Il parametro di uscita AxisFaultId visualizza un valore di errore. Il grafico seguente rappresenta l'errore generato: 
2	Consultare la documentazione CANopen del **Lexium 05** e cercare il codice SigLatched 301C:08.
3	Il valore AxisFaultID è impostato su 4194304. Questo valore binario significa che il bit 22 è impostato su 1. Consultare la documentazione CANopen del **Lexium 05** e cercare il codice SigLatched 301C:08. Il bit 22 per 'SigLatched' designa un errore di lag.
4	Riduce le costanti di velocità nel blocco assoluto, nel carico esterno o nell'accelerazione.
5	Eseguire il blocco `MC_Reset`.

Sostituzione di un servoazionamento guasto

In breve

Se il servoazionamento si guasta, può essere necessario costituirlo con un servoazionamento identico (stesso codice di riferimento prodotto). Per fare questo, si consiglia di salvare i parametri di regolazione in una tabella di dati utilizzando il blocco TE_UPLOADDRIVEPARAMETER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*.

Il blocco TE_DOWNLOADDRIVEPARAM *(vedi pagina 68)* consente di ripristinare in un nuovo servoazionamento i dati salvati.

Backup dei dati

La tabella seguente descrive la procedura per l'esecuzione del backup dei dati del servoazionamento mediante il blocco TE_UPLOADDRIVEPARAMETER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*:

Passo	Azione
1	Disattivare il parametro Enable, che appartiene al blocco MC_POWER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*. **Risultato**: il servoazionamento passa alla modalità Disattiva *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*.
2	Attivare il parametro di ingresso `Execute`. **Risultato**: il servoazionamento passa alla modalità Scaricamento *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*. La tabella di dati assegnata al parametro di uscita `PARAMETERSET` è compilata. **Nota:** eseguire un backup dei dati in un file *.DAT* utilizzando **PLC →Trasferisci dati PLC nel file** se il PLC non dispone di una scheda di memoria.

Ripristino dei dati

La tabella seguente descrive la procedura per il ripristino dei dati del servoazionamento mediante il blocco TE_DOWNLOADDRIVEPARAM *(vedi pagina 68)*:

Passo	Azione
1	Disattivare il parametro Enable, che appartiene al blocco MC_POWER *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*. **Risultato**: il servoazionamento passa alla modalità Disattiva *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*.
2	Sostituire il servoazionamento. Il nuovo servoazionamento deve avere lo stesso codice di riferimento del servoazionamento guasto. **Nota**: accertarsi di adottare tutte le precauzioni necessarie durante la sostituzione del servoazionamento.
3	Configurare il nuovo servoazionamento con i parametri di base *(vedi pagina 49)* (indirizzo CANopen, velocità) oppure utilizzando la tastierina sul pannello frontale.
4	Attivare il parametro di ingresso del blocco `Execute`. **Risultato**: il servoazionamento passa alla modalità Scaricamento *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*. La tabella di dati assegnata al parametro di ingresso `PARAMETERSET` carica il parametro di ingresso `PARAMETERLIST` che corrisponde al parametro del servoazionamento.

Applicazione a più assi

II

Scopo di questa sezione

Questa sezione descrive l'altro hardware disponibile per i blocchi funzione di movimento con un Modicon M340 che esegue Unity Pro.

Il servoazionamento **Lexium 05** è stato usato nella sezione precedente per eseguire un esempio. Questa sezione inizia con una presentazione dei seguenti servoazionamenti in un'architettura completa:

- **Lexium 32**
- **Lexium 15**
- **ATV 31**
- **ATV 32**
- **ATV 71**
- **IcIA**

Dopo questa presentazione viene descritta la configurazione di ogni servoazionamento, specificando le differenze rispetto al **Lexium 05** in modo da eseguire lo stesso esempio.

Contenuto di questa parte

Questa parte contiene i seguenti capitoli:

Capitolo	Titolo del capitolo	Pagina
7	Premessa	89
8	Compatibilità delle applicazioni di movimento con le versioni di Unity Pro	91
9	Implementazione di Lexium 32 per i Blocchi funzione di movimento	93
10	Lexium 15MP/HP/LP Implementazione dei Blocchi funzione di movimento	111
11	ATV 31 Implementazione Blocchi funzione di movimento	135
12	ATV 32 - Implementazione dei Blocchi funzione di movimento (Motion)	151
13	ATV 71 Implementazione dei Blocchi funzione di movimento	165
14	IcIA Implementazione dei Blocchi funzione di movimento	181

Premessa

7

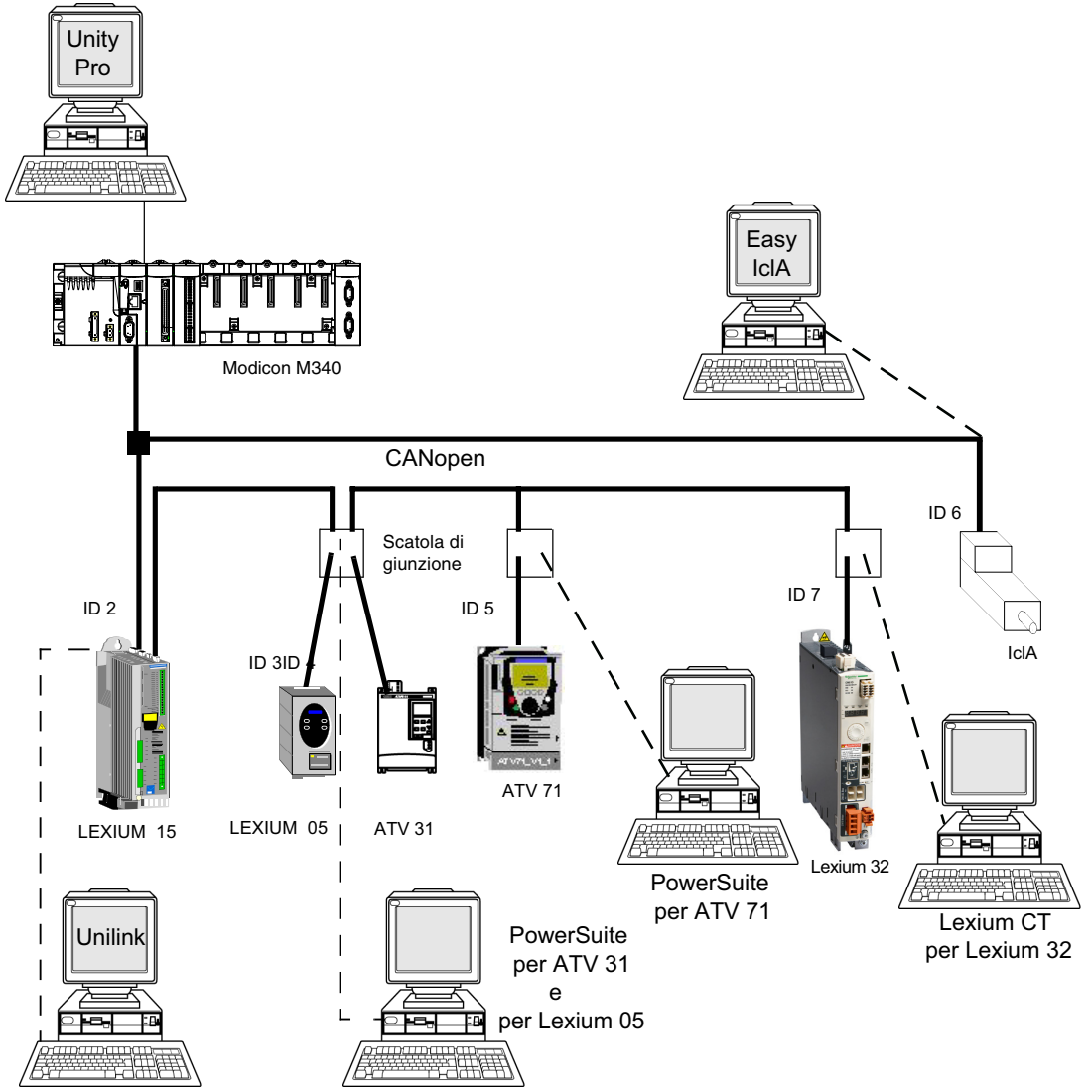
Architettura dell’applicazione con tutti i servoazionamenti

Panoramica

Questa sezione contiene una presentazione dell’uso dell’hardware disponibile (servoazionamenti), mediante un’architettura, per l’implementazione dei Motion Function Blocks in Unity Pro.

Illustrazione

Nella seguente figura è illustrata l'architettura utilizzata nell'applicazione che comprende tutti i servoazionamenti:



Compatibilità delle applicazioni di movimento con le versioni di Unity Pro

8

Compatibilità dei file XEF

Versione Unity di destinazione	Versione Unity sorgente	
	V3.x/V4.0 M340 Proc < V2.0	>= V4.0 M340 Proc >= V2.0
V3.x M340 < V2.0	Compatibilità parziale se si utilizza Lexium15.	NC.
>= V4.0	PC.	FC.
NC: non compatibile. Le parti di movimento vengono ignorate durante l'importazione. PC : compatibilità parziale: i nuovi tipi di asse vengono ignorati con un messaggio di errore durante l'importazione: l'applicazione viene importata dalle sezioni che utilizzano gli azionamenti in errore. Le nuove versioni firmware vengono ridotte alla versione Unity più aggiornata con un avviso durante l'importazione se l'azionamento è presente nel catalogo della CPU Mirano. Altrimenti, l'importazione viene interrotta. FC : compatibile.		

NOTA: 1. : i nuovi EFB provocano errori nelle sezioni che li utilizzano.

NOTA: 2. : Processore M340 >= V2.0 : supporto del salvataggio dei valori iniziali abilitato.

Compatibilità dei file STA

Versione Unity di destinazione	Versione Unity sorgente		
	applicazione V3.x/V4.0 senza motion	V3.x/V4.0 con M340 < V2.0	>= V4.0 con M340 >= V2.0
V3.x	FC	PC	NC
>= V4.0	FC	FC	FC
NC: non compatibile PC : compatibilità parziale: compatibile solo per le applicazioni con azionamento supportato dall'apparecchiatura Unity che apre l'applicazione, nel caso di evoluzione del tipo di azionamento o delle versioni firmware. È possibile aprire l'applicazione, ma non apportare modifiche sostanziali. FC : compatibile.			

Implementazione di Lexium 32 per i Blocchi funzione di movimento

9

Scopo del capitolo

Questa capitolo presenta l'implementazione dei servoazionamenti Lexium 32 conformemente alla metodologia *(vedi pagina 19)* descritta nella guida di avvio rapido *(vedi pagina 13)* con un Lexium 05. Il capitolo specifica in dettaglio solo le differenze e le azioni per un Lexium 32.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
9.1	Adattamento dell'applicazione al Lexium 32	94
9.2	Configurazione del Lexium 32	102
9.3	Regolazione del Lexium 32	105

9.1 Adattamento dell’applicazione al Lexium 32

Scopo di questa sezione

In questa sezione sono descritte le operazioni da eseguire per adattare l’applicazione al **Lexium 32** con un’architettura e sono indicati i requisiti hardware e software.

In questa sezione, Lexium 32 può essere riferito a un Lexium 32 Advanced (LXM 32A...) oppure a un Lexium 32 Modular (LXM 32 M...)

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Architettura dell’applicazione con Lexium 32	95
Requisiti software	96
Requisiti hardware	97
Configurazione bus CANopen del Lexium 32	98

Architettura dell’applicazione con Lexium 32

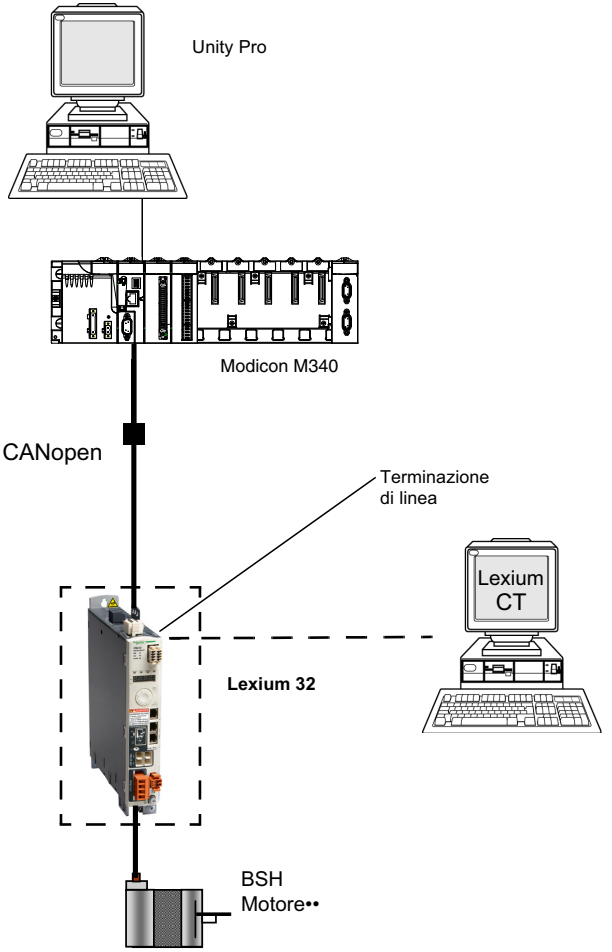
In breve

La struttura proposta rappresenta una struttura semplice che illustra i principi di implementazione del controllo di movimento.

La struttura reale può essere ampliata con altri dispositivi per permettere la gestione di più assi.

Illustrazione

Nella seguente figura è illustrata la struttura utilizzata nell’applicazione:



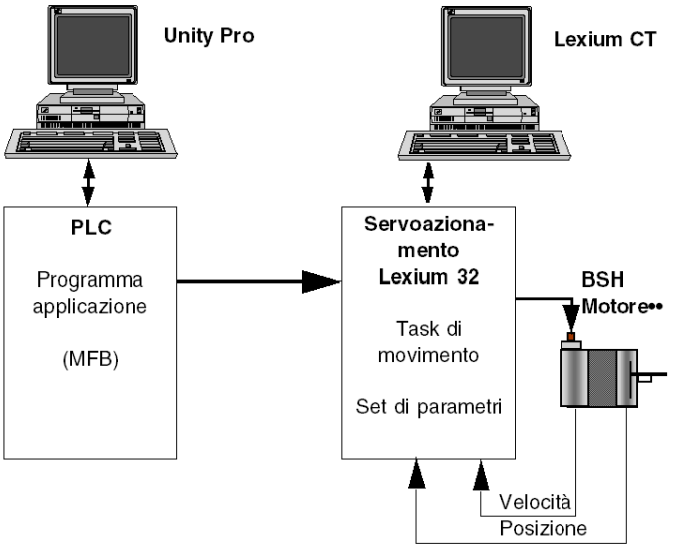
Requisiti software

Panoramica

In conformità ai requisiti software descritti nella Guida di avvio rapido *(vedi pagina 24)*, Lexium CT è utilizzato per la configurazione e la regolazione del **Lexium 32**.

Schema funzionale per il Lexium 32

Il seguente schema illustra le funzioni eseguite dal PLC e dal servoazionamento:



Versioni

Nella seguente tabella sono elencate le versioni hardware e software utilizzate nell'Architettura *(vedi pagina 113)* che consentono l'uso degli MFB in Unity Pro.

Apparecchiatura	Versione software utilizzata nell'esempio	Versione del firmware
Modicon M340	Unity Pro V5.0	>2.0
Lexium 32	Lexium CT V1.0	V1.x for Lexium 32 Advanced V1.y for Lexium 32 Modular

Requisiti hardware

Codici prodotto dei componenti hardware utilizzati

Nella seguente tabella sono elencati i componenti hardware utilizzati nell'architettura *(vedi pagina 95)* che permettono l'implementazione degli MFB **Lexium 32** in Unity Pro.

Hardware	**Riferimento**
PLC **Modicon M340**	**BMX P34 20302**
Alimentatore **Modicon M340**	**BMX CPS 2000**
Rack **Modicon M340**	**BMX XBP 0800**
Lexium 32 Advanced	**LXM32AU90M2**
Cavo di collegamento **Lexium 32** alla porta CANopen del PLC	**TCSCCN4F 3M3T/CAN**
Terminazione di linea CANopen	TCSCAR013M120
Motore per **Lexium 32**	**BSH055••**

Configurazione bus CANopen del Lexium 32

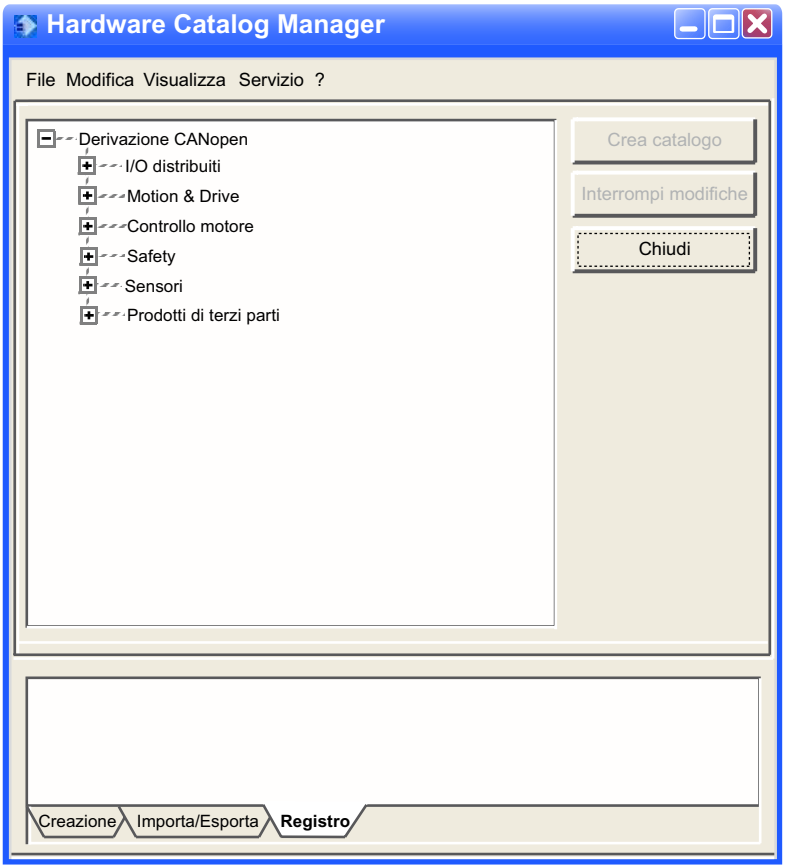
Panoramica

La metodologia per l'implementazione di un bus CANopen tramite Modicon M340 prevede di:

- aggiornare il catalogo hardware
- configurare *(vedi pagina 31)* la porta CANopen per la CPU
- dichiarare lo slave scelto dal catalogo hardware (vedere paragrafo seguente)
- configurare lo slave
- abilitare la configurazione mediante Unity Pro
- verificare *(vedi pagina 35)* il bus CANopen nel browser del progetto.

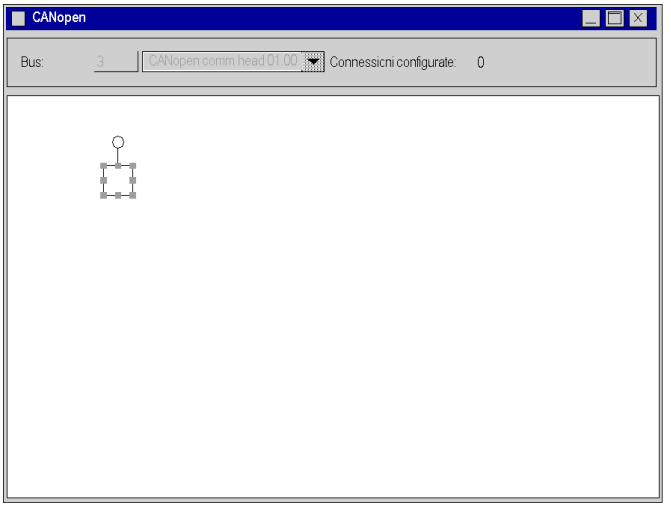
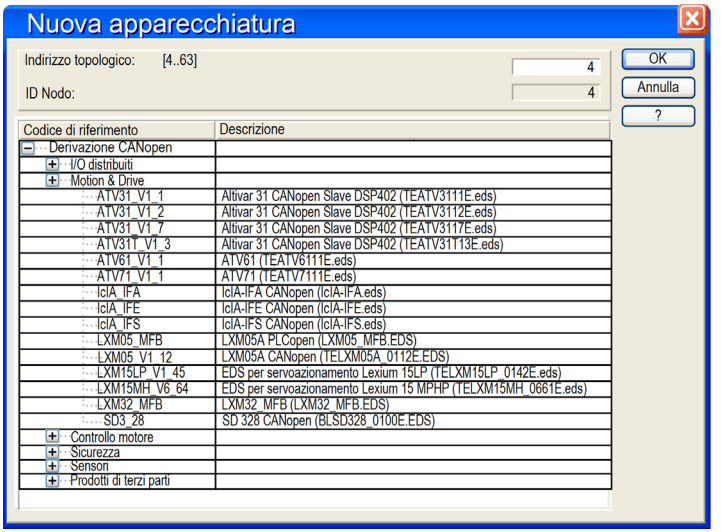
Aggiornamento il catalogo hardware

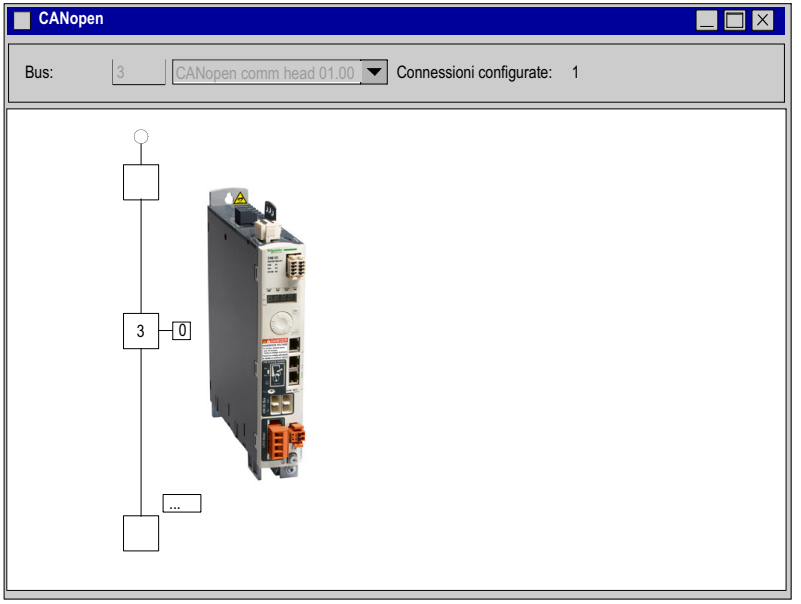
Questa tabella descrive la procedura per configurare lo slave CANopen.

Passo	Azione
1	Aprire il catalogo hardware **Start →Programma →Schneider Electric →Socollaborative →UnityPro →Hardware Catalog Manager** **Risultato:**viene visualizzato Hardware Catalog Manager: 
2	Nella scheda menu, fare clic su File ==>Import User Devices, quindi importare il file LXM32_MFB.cpx nella directory ...\Application Data\Schneider Electric\ConfCatalog\Database\Motion (questo file può trovarsi in una directory nascosta).

Configurazione dello slave CANopen

Questa tabella descrive la procedura per configurare lo slave CANopen.

Passo	Azione
1	Nel **Browser del progetto** di Unity Pro, espandere completamente la directory **Configurazione**, quindi fare doppio clic su **CANopen**. **Risultato:** viene visualizzata la finestra: 
2	Selezionare **Modifica** →**Nuova apparecchiatura**. **Risultato:** viene visualizzata la finestra Nuova apparecchiatura: 

Passo	Azione
3	Impostare 3 in Indirizzo topologico: Come dispositivo slave scegliere Lexium 32.
4	Fare clic su OK per confermare la scelta. **Risultato:** viene visualizzata la finestra CANopen con la nuova apparecchiatura selezionata: 
5	Selezionare **Modifica** →**Apri modulo**. Se MFB non è già selezionato, selezionarlo nell'area Funzione.
6	Il sistema chiede di convalidare le modifiche alla chiusura delle finestre Apparecchiatura e CANopen.

9.2 Configurazione del Lexium 32

Parametri di base per Lexium 32 con Lexium CT

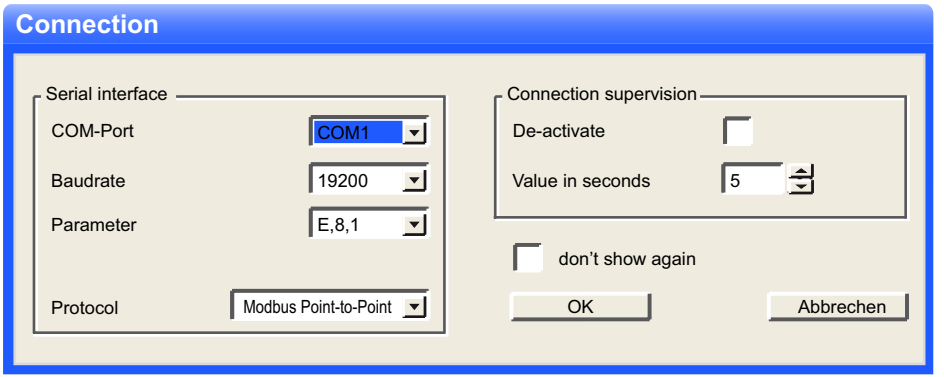
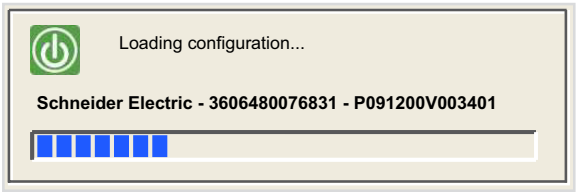
In breve

Lexium CT è un tool di messa in servizio per assi destinato ad applicazioni di controllo del movimento.

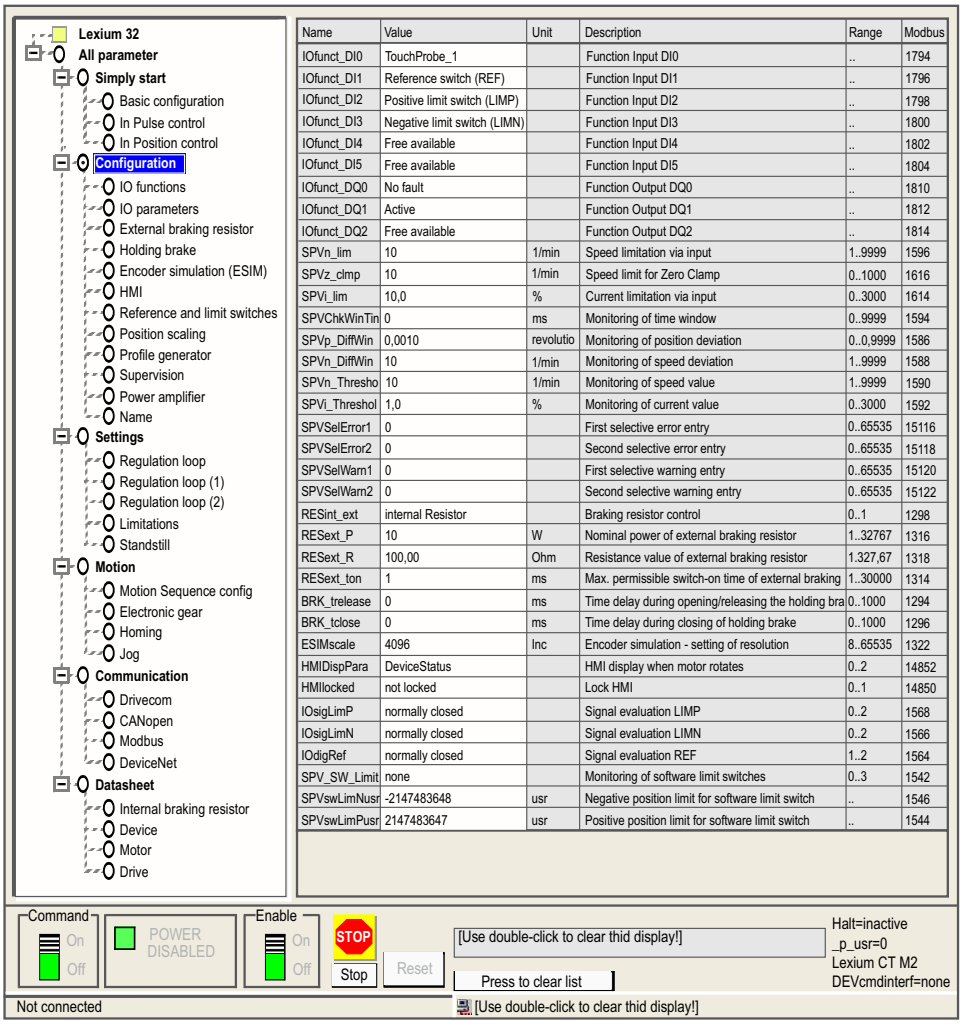
La sua interfaccia utente grafica fornisce un metodo semplice per configurare i parametri di un servoazionamento di tipo **Lexium 32**.

Collegamento a Lexium 32

Questa tabella descrive la procedura di connessione al **Lexium 32.**

Passo	Azione
1	Avviare Lexium CT. Fare clic su **Connection** quindi selezionare il collegamento **ModbusSerialLine connection**. Viene visualizzata la finestra Collegamento:  Selezionare la COM-Port Confermare l'operazione facendo clic su OK. Viene visualizzata le seguente schermata: 

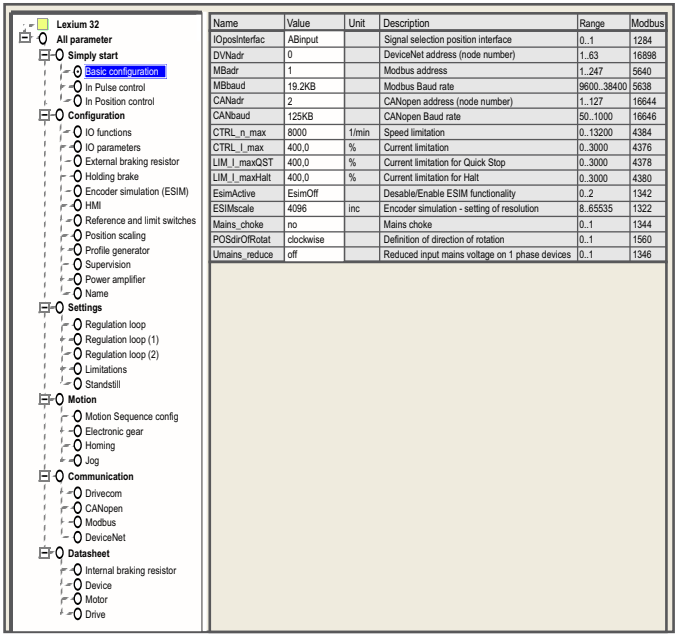
Passo	Azione
2	Dopo che è stata stabilita la configurazione, viene visualizzata questa schermata generale:



Name	Value	Unit	Description	Range	Modbus
IOfunct_DI0	TouchProbe_1		Function Input DI0	..	1794
IOfunct_DI1	Reference switch (REF)		Function Input DI1	..	1796
IOfunct_DI2	Positive limit switch (LIMP)		Function Input DI2	..	1798
IOfunct_DI3	Negative limit switch (LIMN)		Function Input DI3	..	1800
IOfunct_DI4	Free available		Function Input DI4	..	1802
IOfunct_DI5	Free available		Function Input DI5	..	1804
IOfunct_DQ0	No fault		Function Output DQ0	..	1810
IOfunct_DQ1	Active		Function Output DQ1	..	1812
IOfunct_DQ2	Free available		Function Output DQ2	..	1814
SPVn_lim	10	1/min	Speed limitation via input	1..9999	1596
SPVz_clmp	10	1/min	Speed limit for Zero Clamp	0..1000	1616
SPVi_lim	10,0	%	Current limitation via input	0..3000	1614
SPVChkWinTin	0	ms	Monitoring of time window	0..9999	1594
SPVp_DiffWin	0,0010	revolutio	Monitoring of position deviation	0..0,9999	1586
SPVn_DiffWin	10	1/min	Monitoring of speed deviation	1..9999	1588
SPVn_Thresho	10	1/min	Monitoring of speed value	1..9999	1590
SPVi_Threshol	1,0	%	Monitoring of current value	0..3000	1592
SPVSelError1	0		First selective error entry	0..65535	15116
SPVSelError2	0		Second selective error entry	0..65535	15118
SPVSelWarn1	0		First selective warning entry	0..65535	15120
SPVSelWarn2	0		Second selective warning entry	0..65535	15122
RESint_ext	internal Resistor		Braking resistor control	0..1	1298
RESext_P	10	W	Nominal power of external braking resistor	1..32767	1316
RESext_R	100,00	Ohm	Resistance value of external braking resistor	1.327,67	1318
RESext_ton	1	ms	Max. permissible switch-on time of external braking	1..30000	1314
BRK_trelease	0	ms	Time delay during opening/releasing the holding bra	0..1000	1294
BRK_tclose	0	ms	Time delay during closing of holding brake	0..1000	1296
ESIMscale	4096	Inc	Encoder simulation - setting of resolution	8..65535	1322
HMIDispPara	DeviceStatus		HMI display when motor rotates	0..2	14852
HMIlocked	not locked		Lock HMI	0..1	14850
IOsigLimP	normally closed		Signal evaluation LIMP	0..2	1568
IOsigLimN	normally closed		Signal evaluation LIMN	0..2	1566
IOdigRef	normally closed		Signal evaluation REF	1..2	1564
SPV_SW_Limit	none		Monitoring of software limit switches	0..3	1542
SPVswLimNusr	-2147483648	usr	Negative position limit for software limit switch	..	1546
SPVswLimPusr	2147483647	usr	Positive position limit for software limit switch	..	1544

Parametri di base

Questa tabella descrive la procedura per l'immissione dei parametri di base:

Passo	Azione
1	Fare clic sul pulsante **Basic Configuration** Viene visualizzata la finestra **Basic Configuration**:  Name | Value | Unit | Description | Range | Modbus IOposInterfac | ABinput | | Signal selection position interface | 0..1 | 1284 DVNadr | 0 | | DeviceNet address (node number) | 1..63 | 16898 MBadr | 1 | | Modbus address | 1..247 | 5640 MBbaud | 19.2KB | | Modbus Baud rate | 9600..38400 | 5638 CANadr | 2 | | CANopen address (node number) | 1..127 | 16644 CANbaud | 125KB | | CANopen Baud rate | 50..1000 | 16646 CTRL_n_max | 8000 | 1/min | Speed limitation | 0..13200 | 4384 CTRL_I_max | 400,0 | % | Current limitation | 0..3000 | 4376 LIM_I_maxQST | 400,0 | % | Current limitation for Quick Stop | 0..3000 | 4378 LIM_I_maxHalt | 400,0 | % | Current limitation for Halt | 0..3000 | 4380 EsimActive | EsimOff | | Desable/Enable ESIM functionality | 0..2 | 1342 ESIMscale | 4096 | inc | Encoder simulation - setting of resolution | 8..65535 | 1322 Mains_choke | no | | Mains choke | 0..1 | 1344 POSdirOfRotat | clockwise | | Definition of direction of rotation | 0..1 | 1560 Umains_reduce | off | | Reduced input mains voltage on 1 phase devices | 0..1 | 1346 Questa schermata permette di impostare i parametri per l'indirizzo CANopen del servoazionamento, la velocità del bus e le unità usate per accelerazione, velocità e posizione.

2	Per l'esempio di esercitazione, in questa schermata eseguire le seguenti impostazioni o selezioni:
	• Nella zona del servoazionamento:
	  • impostare l'indirizzo CANopen su 2
	  • impostare la velocità di trasmissione del bus a 500 Kbaud *(vedi MFB con Unity Pro, Guida di avvio rapido)*

3	Fare clic su **Elementi** →**Parametro** →**Salva parametri dispositivo** nella EEPROM per confermare **SIMPLYSTART_BASICCONFIGURATION.** **Risultato:** La **SIMPLYSTART_BASICCONFIGURATION** è salvata e viene nuovamente visualizzata la schermata principale.
4	Fare clic su **Esci**.

NOTA: Per informazioni su come regolare i parametri correttamente, fare riferimento alla documentazione dell'azionamento.

9.3 Regolazione del Lexium 32

Scopo di questa sezione

Questa sezione offre un esempio di regolazione del **Lexium 32** con Lexium CT.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Regolazione del Lexium 32	106
Debug del Lexium 32	107

Regolazione del Lexium 32

Modalità di funzionamento

Le modalità di funzionamento disponibili possono essere selezionate nelle schede delle finestre Modalità di funzionamento.

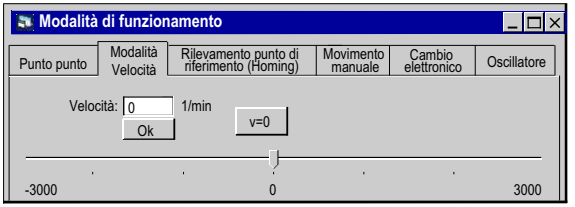
Queste finestre sono suddivise in due sezioni:

- Schede per la modalità operativa selezionata e per l'impostazione dei parametri specifici (sezione superiore)
- Visualizzazione delle informazioni di stato (sezione inferiore)

L'utente può passare da una scheda all'altra della finestra Modalità di funzionamento senza interferire con la modalità di funzionamento correntemente attiva.

Profilo di velocità

Nella modalità di funzionamento Profilo di velocità, l'azionamento accelera fino a una velocità di destinazione regolabile. È possibile impostare un profilo di movimento con valori per le rampe di accelerazione e decelerazione



Debug del Lexium 32

Prerequisito

Si consiglia di eseguire il debug della dinamica dell'asse prima che venga avviato automaticamente dal programma.

Descrizione

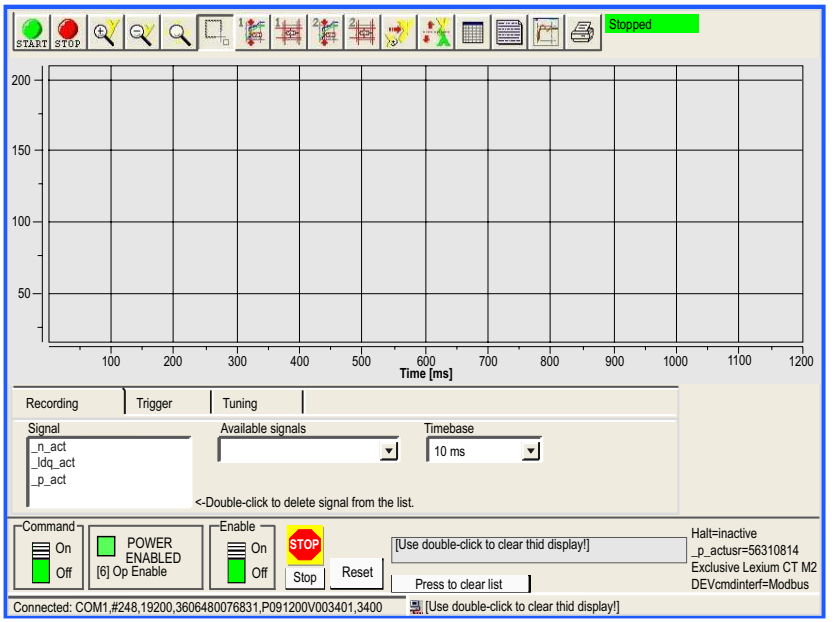
Il software di messa in servizio fornisce la funzione "**Registrazione / Regolazione**" per la visualizzazione dei dati interni del dispositivo durante i movimenti. Il dispositivo collegato memorizza i dati di movimento in una memoria interna per un periodo di registrazione definito, quindi li invia al PC. Il PC elabora i dati e li visualizza sotto forma di grafici o tabelle.

I dati registrati possono essere salvati nel PC e possono essere archiviati o stampati per essere inclusi nella documentazione.

Utilizzare la **voce di menu " →Funzioni →Registrazione / Regolazione..."** per avviare la funzione "registrazione".

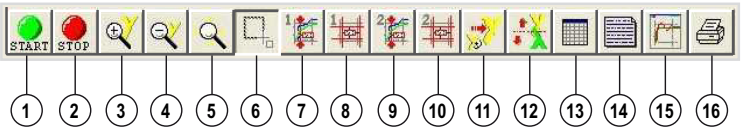
Illustrazione

Per accedere alla seguente schermata, fare clic sulla scheda **Oscilloscopio**:



Descrizione dei pulsanti

È possibile accedere ai seguenti pulsanti facendo clic su:

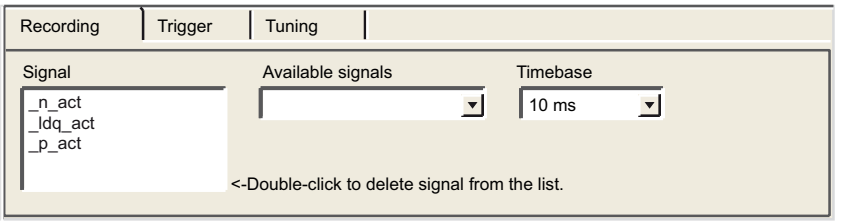


1. Avvio registrazione
2. Arresto registrazione
3. Ingrandimento, asse y
4. Riduzione, asse y
5. Zoom variabile all'infinito, asse x e asse y
6. Ingrandimento/riduzione del rettangolo selezionato.
7. Prima visualizzazione dei valori per un periodo specifico
8. Modifica dei valori visualizzati per la prima visualizzazione
9. Seconda visualizzazione dei valori per un periodo specifico
10. Modifica dei valori visualizzati per il secondo visualizzazione
11. Ripristino della visualizzazione originale
12. Inversione dell'asse y
13. Visualizzazione della tabella di valori registrati
14. Immissione della descrizione
15. Mostra/nasconde la configurazione
16. Stampa la registrazione

Recording

I parametri desiderati vengono selezionati nel campo di immissione "Segnali disponibili". È possibile selezionare al massimo 4 parametri. Se un parametro non è più disponibile, è possibile deselezionarlo facendo doppio clic sul nome del parametro.

L'incremento di registrazione desiderato viene selezionato nel campo di immissione "Base tempo". Più piccola è la "Base tempo", più breve sarà il massimo tempo di registrazione.

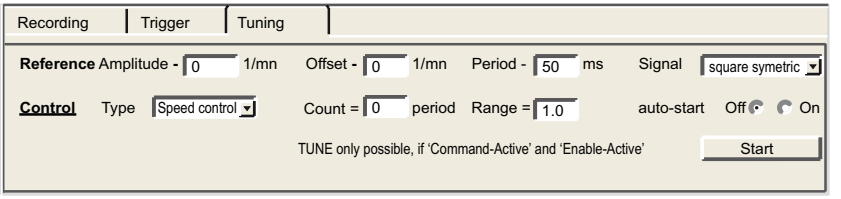


Tuning

La regolazione può essere avviata solo se i commutatori "Access" e "Enable" sono impostati a "On".

- Il campo "Ampiezza" permette di impostare l'ampiezza massima del valore di riferimento
- L'offset dell'ampiezza in direzione negativa e positiva può essere impostato nel campo "Offset".
- La durata di un periodo viene impostata nel campo "Periodo".
- Il tipo di segnale per il valore di riferimento viene impostato nell'elenco a discesa "Segnale".
- Il controller da utilizzare viene impostato nell'elenco a discesa "Tipo".
- Il numero di periodi è specificato nel campo "Cont".
- Il numero massimo di giri che possono essere attivati con la regolazione può essere impostato nel campo "Intervallo". Questo valore, ad esempio, può contribuire ad evitare il blocco di un movimento.
- I pulsanti di opzione "avvio automatico" permettono di collegare l'esecuzione del movimento di regolazione e l'inizio della registrazione. Se l'opzione è impostata a "Off", il software visualizza il pulsante Avvio. Il pulsante Avvio consente di innescare il movimento di regolazione indipendentemente dall'avvio della registrazione.

NOTA: le impostazioni effettuate nella scheda "Trigger" vanno perse se si imposta "avvio automatico" a "On".



NOTA: per maggiori informazioni fare riferimento al manuale utente del software Lexium CT.

Lexium 15MP/HP/LP Implementazione dei Blocchi funzione di movimento

10

Scopo di questo capitolo

Questa capitolo presenta l'implementazione di un servoazionamento Lexium 15MP/HP/LP conformemente alla metodologia *(vedi pagina 19)* descritta nella guida di avvio rapido *(vedi pagina 13)* con un Lexium 05. Il capitolo specifica in dettaglio solo le differenze e le azioni per un Lexium 15MP/HP/LP.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
10.1	Adeguamento dell'applicazione agli azionamenti Lexium 15MP/HP/LP	112
10.2	Configurazione del bus CANopen per gli azionamenti Lexium 15MP/HP/LP	117
10.3	configurazione del Lexium 15MP/HP/LP	120
10.4	regolazione del Lexium 15MP/HP/LP	131

10.1 Adeguamento dell’applicazione agli azionamenti Lexium 15MP/HP/LP

Scopo di questa sezione

In questa sezione sono descritte le operazioni da eseguire per adeguare l’applicazione agli azionamenti **Lexium 15MP/HP/LP** con un’architettura e sono indicati i requisiti hardware e software.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Architettura di un’applicazione con gli azionamenti Lexium 15MP/HP/LP	113
Requisiti software	114
Requisiti hardware	116

Architettura di un’applicazione con gli azionamenti Lexium 15MP/HP/LP

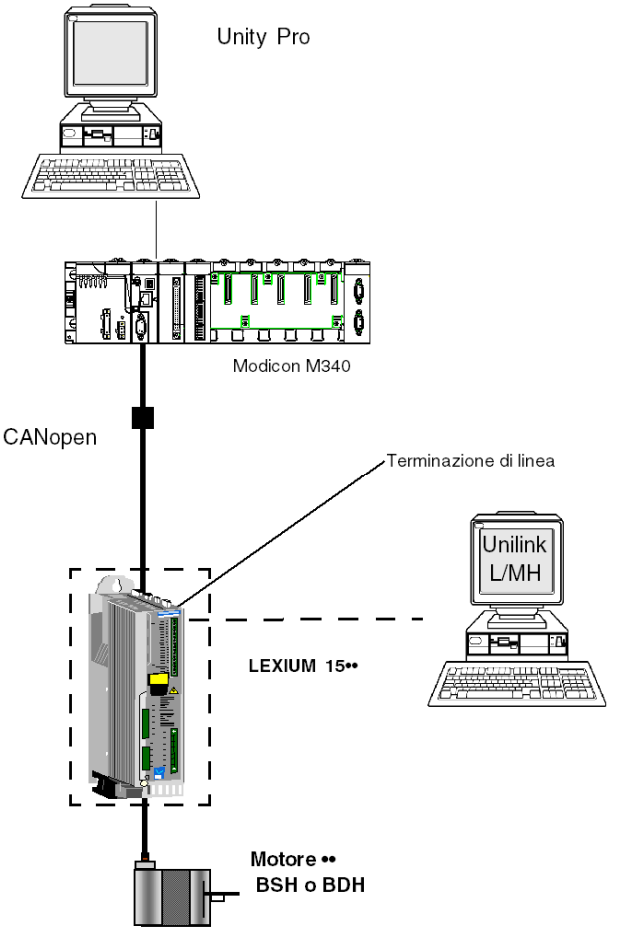
In breve

La struttura proposta rappresenta una struttura semplice che integra i principi di implementazione del controllo di movimento.

La struttura reale può essere ampliata con altri dispositivi per permettere la gestione di più assi.

Illustrazione

Nella seguente figura è illustrata la struttura utilizzata nell’applicazione:



Requisiti software

Panoramica

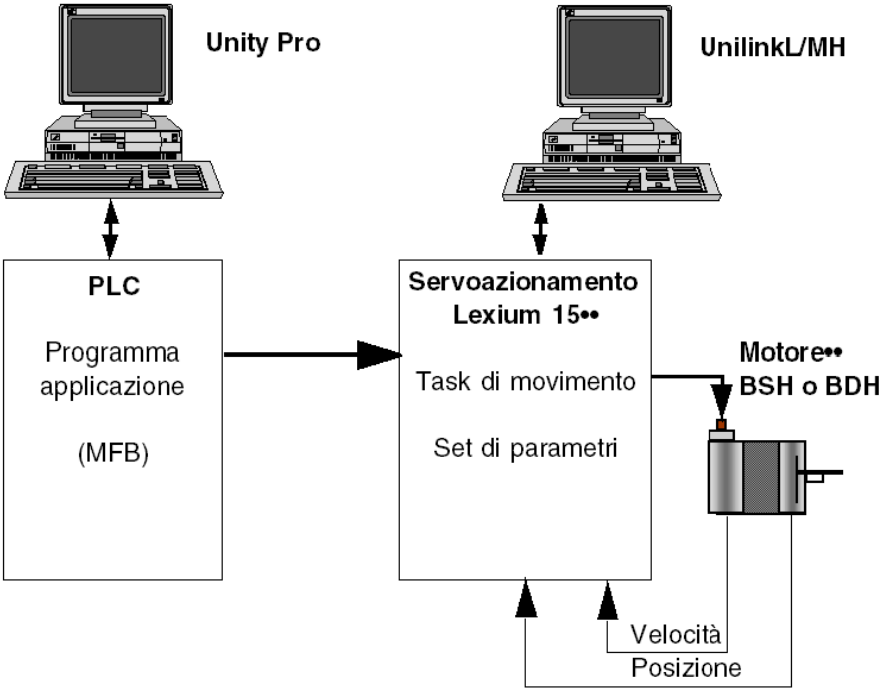
In riferimento ai requisiti software indicati nella Guida di avvio rapido *(vedi pagina 24)*, PowerSuite è utilizzato per la configurazione e la regolazione del **Lexium 05**.

PowerSuite for **Lexium 05** consente la regolazione degli assi e offre un metodo semplice per configurare i parametri di un servoazionamento **Lexium 05**.

Unilink L/MH for **Lexium 15••** svolge le stesse funzioni, ma per un azionamento **Lexium 15••**.

Schema funzionale per il Lexium 15••

Il seguente schema illustra le funzioni eseguite dal PLC e dal servoazionamento:



Versioni

Nella seguente tabella sono elencate le versioni hardware e software utilizzate nell'architettura *(vedi pagina 113)* che consentono l'uso degli MFB in Unity Pro.

Apparecchiatura	Versione software utilizzata nell'esempio	Versione del firmware
Modicon M340	Unity Pro V4.0	-
Lexium 15LP	Unilink V1.5	V1.45 solo funzione MFB V2.36 gestita da MTM
Lexium 15MH	Unilink V4.0	Compatibile dalla V6.64

Requisiti hardware

Codici prodotto dei componenti hardware utilizzati

Nella seguente tabella sono elencati i componenti hardware utilizzati nell'architettura *(vedi pagina 113)* che permettono l'implementazione degli MFB **Lexium 15MP** in Unity Pro.

Hardware	Riferimento
PLC **Modicon M340**	**BMX P34 2030**
Alimentatore **Modicon M340**	**BMX CPS 2000**
Rack **Modicon M340**	**BMX XBP 0800**
Servoazionamento **Lexium 15MP**	**LXM15MD28N4**
Cavo di collegamento **Lexium 15MP** alla porta CANopen del PLC	**TLA CD CBA •••**
Connettore CANopen per **Lexium 15MP**	**AM0 2CA 001 V000**
Motore per **Lexium 15MP**	**BPH055••**

Nella seguente tabella sono elencati i componenti hardware utilizzati nell'architettura *(vedi pagina 113)* che permettono l'implementazione degli MFB **Lexium 15LP** in Unity Pro.

Hardware	Riferimento
PLC **Modicon M340**	**BMX P34 2030**
Alimentatore **Modicon M340**	**BMX CPS 2000**
Rack **Modicon M340**	**BMX XBP 0800**
Servoazionamento **Lexium 15LP**	**LXM15LD13M3**
Cavo di collegamento **Lexium 15MP** alla porta CANopen del PLC	**TLA CD CBA •••**
Connettore CANopen per **Lexium 15LP**	**AM0 2CA 001 V000**
Motore per **Lexium 15LP**	**AKM 31E**

NOTA: Il terminatore di linea è un sezionatore integrato nel connettore CANopen **AM0 2CA 001 V000**.

10.2 Configurazione del bus CANopen per gli azionamenti Lexium 15MP/HP/LP

Configurazione dello slave CANopen sul bus CANopen

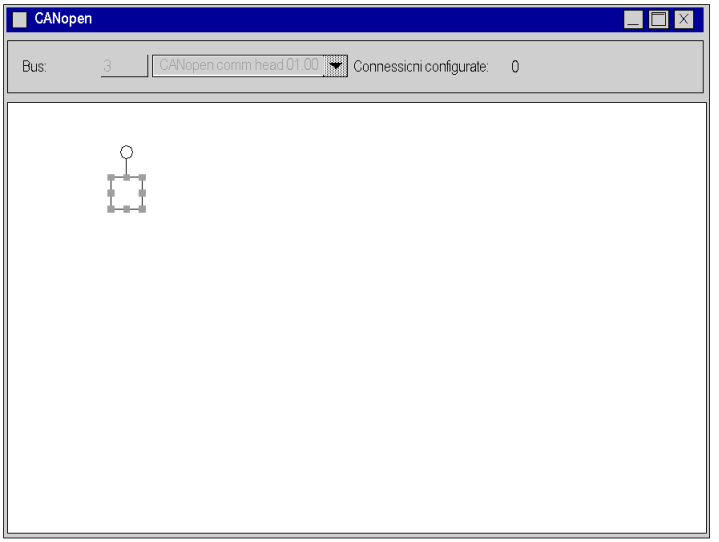
Panoramica

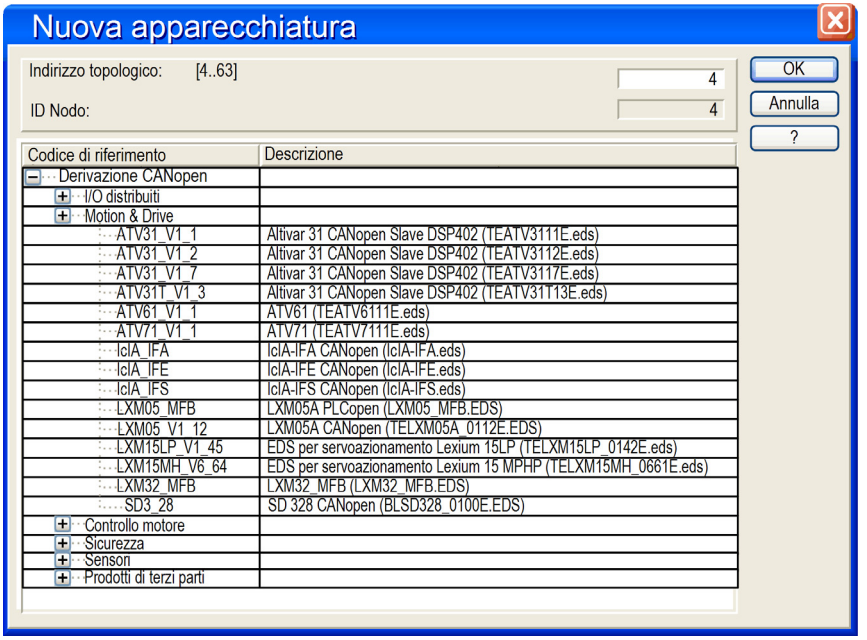
La metodologia per l'implementazione di un bus CANopen tramite Modicon M340 prevede di:

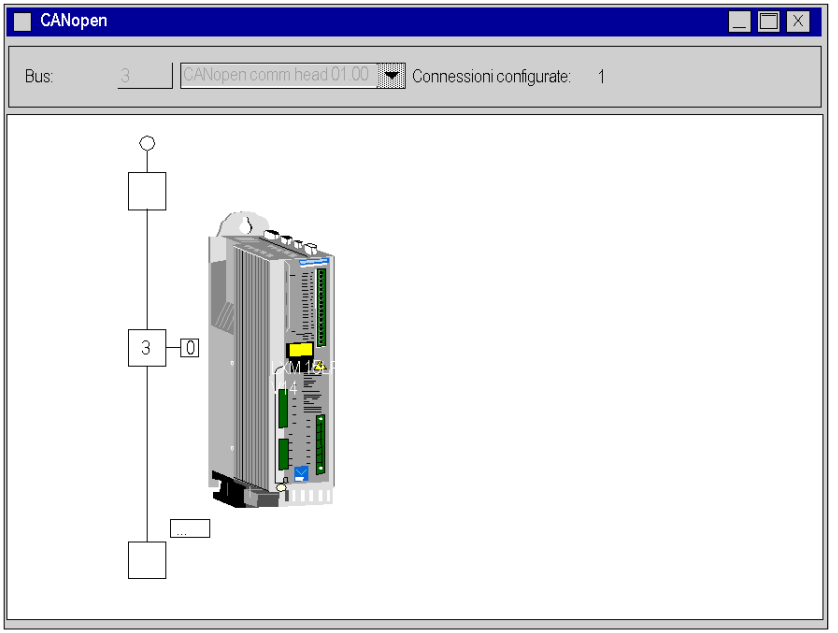
- configurare *(vedi pagina 31)* la porta CANopen per la CPU
- dichiarare lo slave scelto dal catalogo hardware (vedere paragrafo seguente)
- configurare lo slave
- attivare la configurazione mediante Unity Pro
- verificare *(vedi pagina 35)* il bus CANopen nel browser del progetto.

Configurazione dello slave CANopen

Questa tabella descrive la procedura per configurare lo slave CANopen.

Procedura	Azione
1	Nel **Browser del progetto** di Unity Pro, espandere completamente la directory **Configurazione**, quindi fare doppio clic su **CANopen**. **Risultato:** viene visualizzata la finestra: 

Procedura	Azione
2	Selezionare **Modifica** →**Nuova apparecchiatura**. **Risultato:** viene visualizzata la finestra Nuova apparecchiatura: 
3	Impostare 3 in Indirizzo topologico: per il dispositivo slave selezionare Lexium15LP_V1_4, per un Lexium 15LP, o Lexium15MH_V6_61 per un Lexium15MP.

Procedura	Azione
4	Fare clic su OK per confermare la scelta. **Risultato:** viene visualizzata la finestra CANopen con la nuova apparecchiatura selezionata: 
5	Selezionare **Modifica** →**Apri modulo**. Se MFB non è già selezionato, selezionarlo nell'area Funzione.
6	Il sistema chiede di convalidare le modifiche alla chiusura delle finestre Apparecchiatura e CANopen.

10.3 configurazione del Lexium 15MP/HP/LP

Scopo di questa sezione

In questa sezione sono descritte le configurazioni di base dei servoazionamenti mediante **Unilink L/MH** per **Lexium 15MP/HP/LP**.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Parametri di base per Lexium 15MP tramite Unilink MH	121
Parametri di base per Lexium 15LP tramite Unilink L	124
Parametri specifici per Lexium 15 MP/HP/LP con Unilink	129

Parametri di base per Lexium 15MP tramite Unilink MH

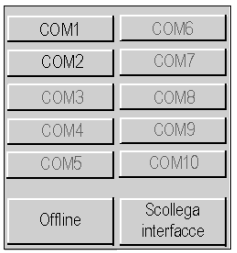
In breve

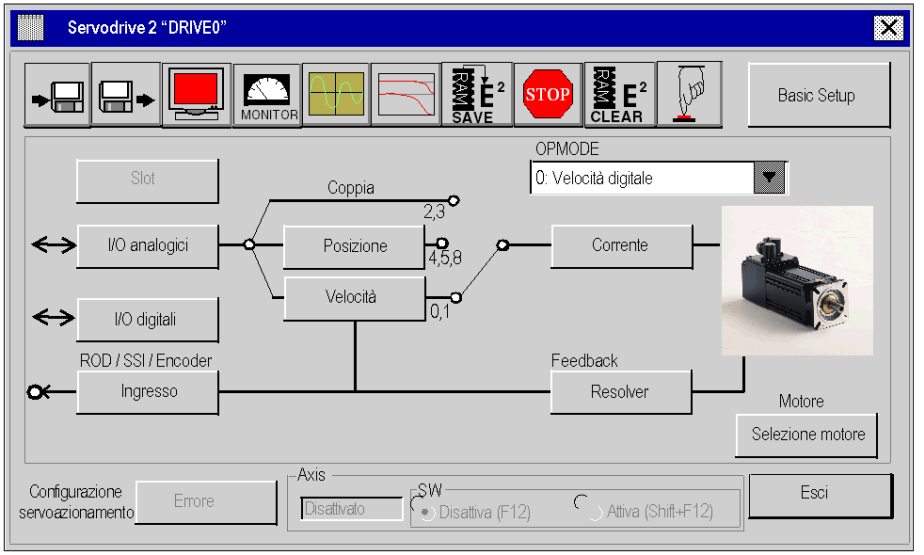
Unilink è un tool di messa in servizio per assi destinato ad applicazioni di controllo del movimento.

La sua interfaccia utente grafica fornisce un metodo semplice per configurare i parametri di un servoazionamento di tipo **Lexium 15MP**.

Collegamento a Lexium 15MP

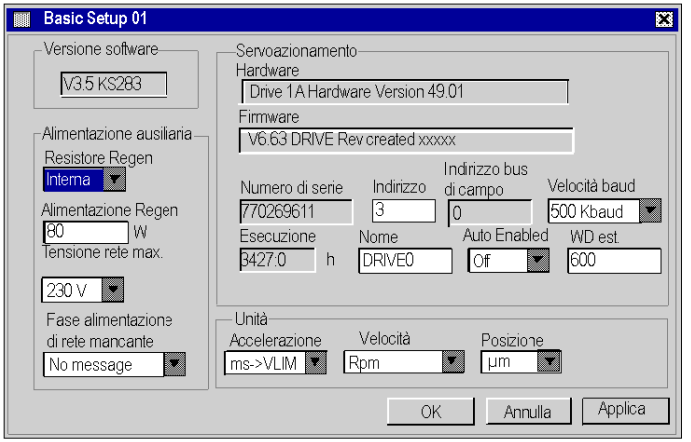
Questa tabella descrive la procedura di connessione al servoazionamento **Lexium 15MP**:

Passo	Azione
1	Avviare Unilink MH selezionando **Avvio** →**Programma** →**Unilink** →**Unilink MH**. Sulla schermata principale di Unilink MH viene visualizzata una finestra di comunicazione:  Se la porta che si sta utilizzando è disponibile (ovvero non è usata da altri dispositivi o programmi), il nome COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM 8, COM9, COM10 è visualizzato in nero. In caso contrario è visualizzato in grigio.

Passo	Azione
2	Fare clic su una di queste porte di comunicazione (la porta che si usa sul PC) per trasferire i valori dei parametri del servozionamento al PC. Dopo che è stata stabilita la comunicazione, viene visualizzata questa schermata generale: 

Parametri di base

Questa tabella descrive la procedura per l'immissione dei parametri di base:

Passo	Azione
1	Fare clic sul pulsante **Configurazione di base** nella schermata generale. Viene visualizzata la finestra **Configurazione di base**:  Questa schermata permette di impostare i parametri per l'indirizzo CANopen del servoazionamento, la velocità del bus e le unità usate per accelerazione, velocità e posizione.

2	Per l'esempio di esercitazione, in questa schermata eseguire le seguenti impostazioni o selezioni:
	• Nella zona del servoazionamento:
	  • impostare l'indirizzo CANopen su 2
	  • impostare la velocità di trasmissione del bus a 500 Kbaud *(vedi pagina 31)*
	• Nella zona Unit *(vedi Unity Pro, Blocchi funzione Motion, Libreria dei blocchi funzione)*:
	  • l'accelerazione in ms->VLIM
	  • la velocità in rpm
	  • la posizione in µm

3	Fare clic sui pulsanti **Selezione motore**, **Corrente**, **Resolver** per dichiarare il motore e i parametri di feedback. **Nota:** per informazioni su come dichiarare correttamente il motore, fare riferimento alla documentazione del motore.
4	Fare clic su **OK** per confermare la configurazione di base. **Risultato:** la configurazione di base è salvata e viene nuovamente visualizzata la schermata principale. **Nota:** quando alcuni parametri ASCII sono stati attivati, compare una finestra che chiede di salvare le modifiche alla memoria EEPROM del servoazionamento. Fare clic su **OK** per riavviare il servoazionamento e aggiornare la memoria.
5	Fare clic su **Esci**.

Parametri di base per Lexium 15LP tramite Unilink L

In breve

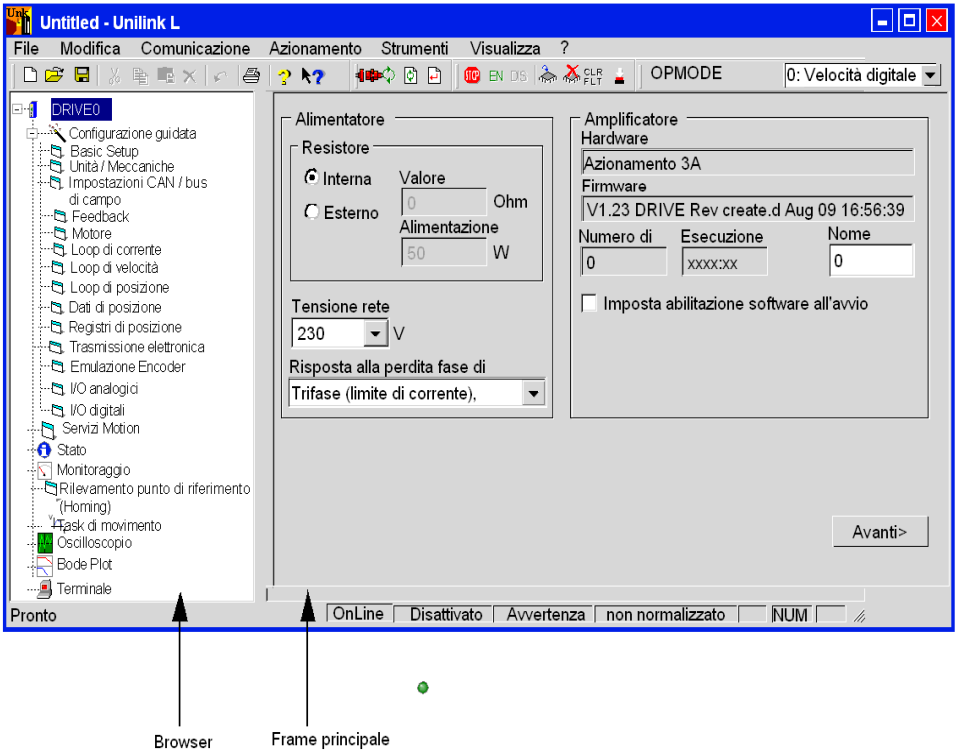
Unilink è un tool di messa in servizio per assi destinato ad applicazioni di controllo del movimento.

La sua interfaccia utente grafica fornisce un metodo semplice per configurare i parametri di un servoazionamento di tipo **Lexium 15LP**.

Collegamento al Lexium 15LP

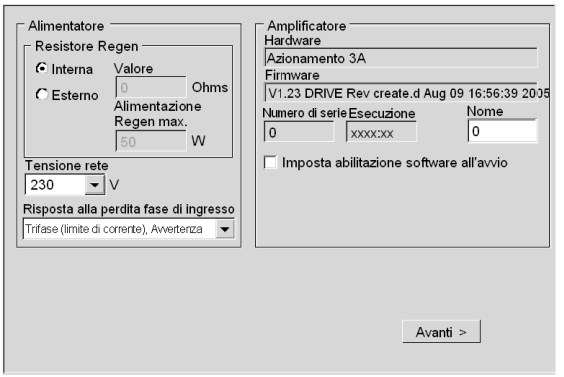
Questa tabella descrive la procedura di connessione al **Lexium 15LP**:

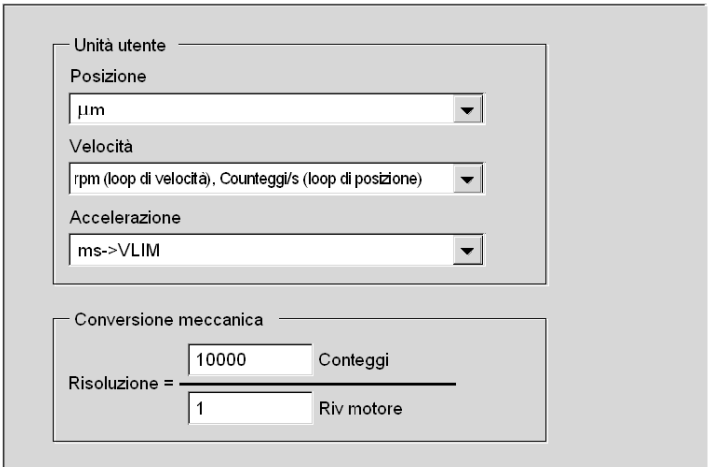
Passo	Azione
1	Avviare Unilink L selezionando **Avvio** →**Programma** →**Unilink** →**Unilink L**. **Risultato**: una finestra chiede se si desidera collegarsi all'azionamento.
2	Fare clic sul pulsante **Sì**. **Risultato**: viene visualizzata una finestra per selezionare il dispositivo.
3	Selezionare **RS-232** e fare clic sul pulsante **OK**. **Risultato**: viene visualizzata una finestra delle impostazioni RS-232.
4	Impostare la porta seriale (da COM1 a COM10), la velocità di trasmissione (38400), il timeout (2000ms).

Passo	Azione
5	Fare clic sul pulsante **OK**. **Risultato**: viene visualizzato il software Unilink L. 

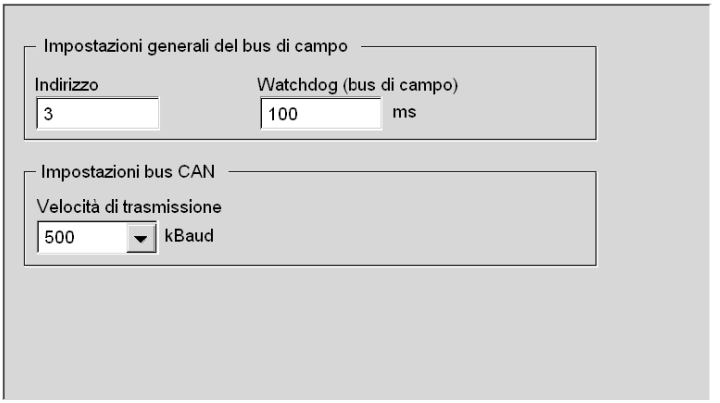
Parametri di base

Questa tabella descrive la procedura per l'immissione dei parametri di base:

Passo	Azione
1	Fare clic su **Configurazione guidata** nel browser. **Risultato**: viene visualizzata la schermata **Configurazione azionamento** nel frame principale:
2	Nella schermata selezionare **Configurazione completa**. **Risultato**: viene visualizzato il browser con tutti i collegamenti di configurazione.
3	Fare clic su **Configurazione di base** nel browser. **Risultato**: viene visualizzata la schermata **Configurazione di base** nel frame principale:  Questa schermata permette di impostare i parametri di alimentazione.

Passo	Azione
4	Fare clic su **Unità/Meccaniche** nel browser. Viene visualizzata la schermata **Unità/Meccaniche** nel frame principale:  Per l'esempio di esercitazione, in questa schermata eseguire le seguenti impostazioni o selezioni: ● Nella zona Unità utente: ● l'accelerazione in ms->VLIM ● la velocità in rpm ● la posizione in µm

Passo	Azione

5	Fare clic su **Impostazioni CAN / bus di campo** nel browser.
	Viene visualizzata la schermata **Impostazioni CAN / bus di campo** nel frame principale:
	 
	
	Per l'esempio di esercitazione, in questa schermata eseguire le seguenti impostazioni o selezioni:
	● Nelle zone Impostazioni generali del bus di campo e del bus CAN:
	  ● impostare l'indirizzo CANopen su 3
	  ● impostare la velocità di trasmissione del bus a 500 Kbaud

6	Fare clic sulle cartelle **Motore**, **Resolver** nel browser per dichiarare i parametri del motore e di feedback. **Nota:** per informazioni su come dichiarare correttamente il motore, fare riferimento alla documentazione del motore.
7	Salvare i parametri tramite **Azionamento** →**Salva nella EEPROM**. **Risultato:** la configurazione di base è salvata e viene nuovamente visualizzata la schermata principale.

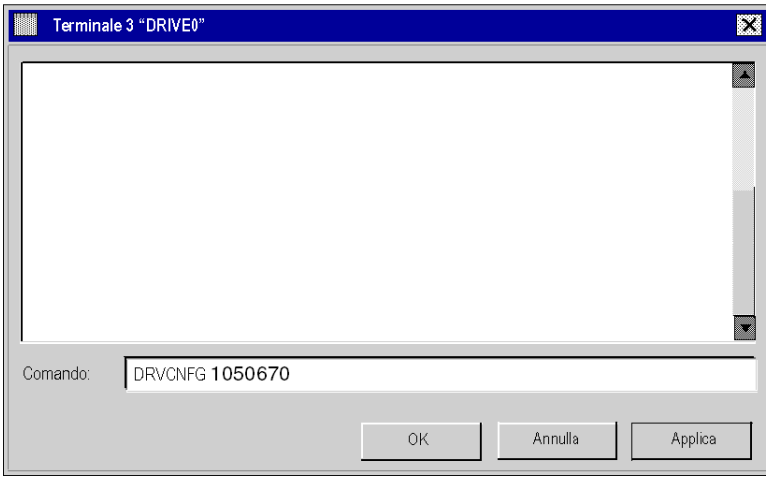
Parametri specifici per Lexium 15 MP/HP/LP con Unilink

In breve

In aggiunta ai parametri di base *(vedi MFB con Unity Pro, Guida di avvio rapido)* vengono immessi dei parametri specifici. Questi parametri specifici integrano la configurazione di **Lexium 15 MP/HP/LP** modificando alcuni codici ASCII mediante la finestra **Terminale**.

Parametri specifici

Nella seguente tabella è descritta la procedura per l'immissione di parametri specifici del **Lexium 15 MP/HP/LP**:

Passo	Azione
1	Fare clic sull'icona **Terminale** nella pagina generale. Viene visualizzata la finestra **Terminale**:  In questa schermata è possibile configurare il punto di connessione di un **Lexium 15MP/HP/LP**.
2	Per Lexium 15 MP/HP, immettere nel campo **Comando**: ● DRVCNFG 1050670 Per Lexium 15 LP, immettere nel campo **Comando**: ● INPT2 x1,5 durata del task oppure IN20Mode42 **MAST** o **FAST**
3	Fare clic su **Applica** per confermare la configurazione di questo parametro ASCII.

Passo	Azione
4	Per Lexium 15 MP/HP, ripetere i passi indicati immettendo nel campo **Comando**: ● DRVCNFG2 64 ● INPT x1,5 durata del task **MAST** o **FAST** ● ENGAGE 1
5	Fare clic su **OK** per confermare l'ultimo **Comando** e ritornare alla pagina generale.
6	Fare clic sull'icona **Salva** nella pagina generale per salvare i parametri di base e quelli specifici nella memoria EEPROM del servoazionamento.
7	Chiudere la finestra generale e fare clic su **DIS** per scollegarsi dal servoazionamento.

Comando

Immettere qui il comando ASCII, con i parametri corrispondenti. Confermare l'immissione premendo **INVIO** o premere il pulsante **APPLICA** per avviare la trasmissione.

⚠ ATTENZIONE

FUNZIONAMENTO ANOMALO DELL'APPLICAZIONE

Prima di emettere il comando ASCII, verificare che sia appropriato per l'apparecchiatura.

Il mancato rispetto di queste istruzioni può provocare infortuni o danni alle apparecchiature.

10.4 regolazione del Lexium 15MP/HP/LP

Debug dell’asse

Prerequisito

Si consiglia di eseguire il debug della dinamica dell’asse prima che venga avviato automaticamente dal programma.

Descrizione

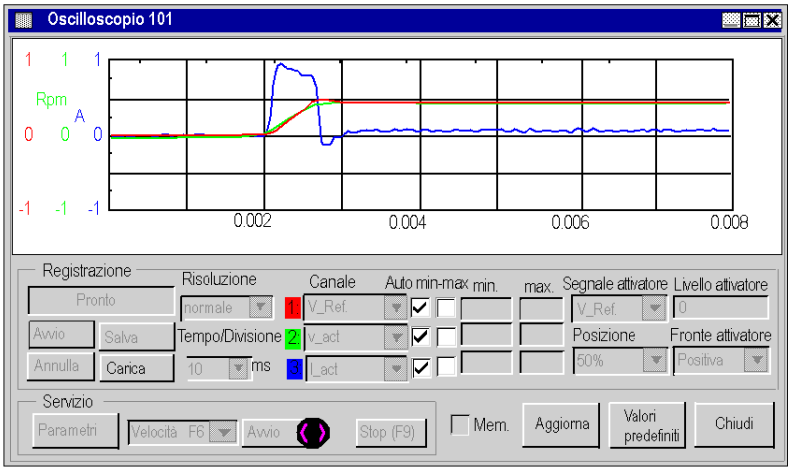
L’operazione di debug può essere effettuata con l’oscilloscopio.

Questo permette di:

- visualizzare fino a tre variabili simultaneamente, come una funzione di tempo
- salvare le misure registrate su un supporto dati in formato CSV (può essere usato con MS-Excel)
- caricare un file di dati CSV e ripristinare le curve sul diagramma dell’oscilloscopio
- utilizzare determinati servizi

Illustrazione per Lexium 15MH

La schermata seguente può essere visualizzata facendo clic sul menu **Unilink MH Strumenti →Oscilloscopio**:



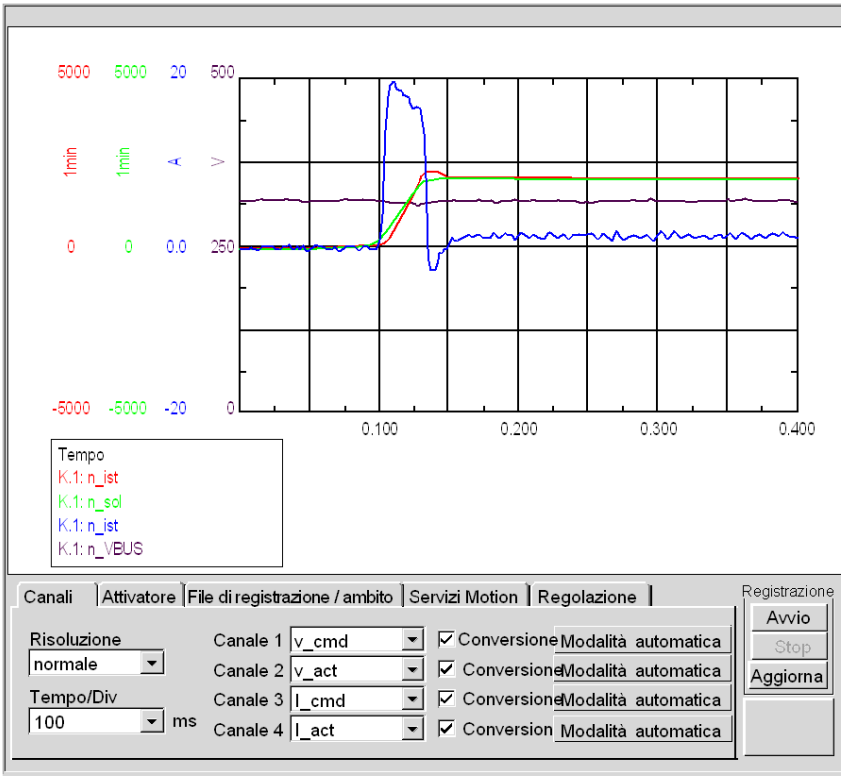
Avvio di un servizio per Lexium 15MH

La tabella seguente spiega come utilizzare una funzione di servizio con un Lexium 15MH:

Passo	Azione
1	Nel campo **Servizio**, selezionare una delle funzioni di servizio *(vedi pagina 133)* descritte di seguito.
2	Fare clic sul pulsante **Parametri**.
3	Impostare il parametro corrispondente.
4	Quindi avviare la funzione selezionando il pulsante **Avvio**.
5	La funzione continuerà ad essere eseguita fino a quando non si fa clic sul pulsante **Stop** o non si preme il tasto funzione **F9**.

Illustrazione per Lexium 15LP

La schermata seguente può essere visualizzata facendo clic sulla cartella **Oscilloscopio** nel browser **Unilink L**:



Avvio di un servizio per Lexium 15LP

La tabella seguente spiega come utilizzare una funzione di servizio con un Lexium 15LP:

Passo	Azione
1	Fare clic sulla scheda **Servizi di movimento**.
2	Selezionare una delle funzioni di servizio *(vedi pagina 133)* descritte di seguito.
3	Fare clic sul pulsante **Parametri**.
4	Impostare il parametro corrispondente.
5	Quindi avviare la funzione selezionando il pulsante **Avvio**.
6	La funzione continuerà ad essere eseguita fino a quando non si fa clic sul pulsante **Stop**.

Funzioni di servizio

La tabella seguente spiega come utilizzare una funzione di servizio:

Corrente continua	Applicare una corrente continua al motore con dimensioni regolabili e angolo del vettore di campo elettrico. Il passaggio dal controllo di velocità al controllo di corrente viene eseguito automaticamente, la commutazione avviene indipendentemente dal feedback (resolver o simile). Il rotore si blocca su un polo dello statore.
Velocità	Fa funzionare l'azionamento a velocità costante. Viene fornito un setpoint digitale interno (la velocità è regolabile).
Coppia	Fa funzionare l'azionamento con corrente costante. Viene fornito un setpoint digitale interno (la corrente è regolabile). Il passaggio dal controllo di velocità al controllo di corrente viene eseguito automaticamente, la commutazione avviene indipendentemente dal feedback (resolver o simile).
Modalità reversing	Fa funzionare l'azionamento in modalità reversing, con velocità regolabile separatamente e tempo di inversione per ogni direzione di rotazione.
Task di movimento	Avvia il task di movimento che viene selezionato nella schermata "Immissione di parametri di servizio".
Zero	Funzione usata per l'impostazione del feedback in combinazione con la fase di posizionamento. A questa funzione è possibile accedere solo in OMODE2.

NOTA: per maggiori informazioni fare riferimento al manuale utente del software Unilink.

NOTA: una volta impostati correttamente i parametri, si consiglia di salvarli nella EEPROM e di eseguirne una copia di backup in un file.

ATV 31 Implementazione Blocchi funzione di movimento

11

Scopo di questo capitolo

Questa capitolo presenta l'implementazione di un servoazionamento ATV 31 conformemente alla metodologia *(vedi pagina 19)* descritta nella guida di avvio rapido *(vedi pagina 13)* con un Lexium 05. Il capitolo specifica in dettaglio solo le differenze e le azioni per un ATV 31.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
11.1	Adattamento dell'applicazione all'ATV 31	136
11.2	Configurazione del bus CANopen ATV 31	140
11.3	Configurazione del ATV 31	143
11.4	Regolazione dell'ATV 31	149

11.1 Adattamento dell’applicazione all’ATV 31

Obiettivo di questa sezione

Questa sezione spiega come adattare l’applicazione all’**ATV 31** con un tipo di architettura e indica i requisiti hardware e software.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Architettura di un’applicazione con un ATV 31	137
Requisiti software	138
Requisiti hardware	139

Architettura di un’applicazione con un ATV 31

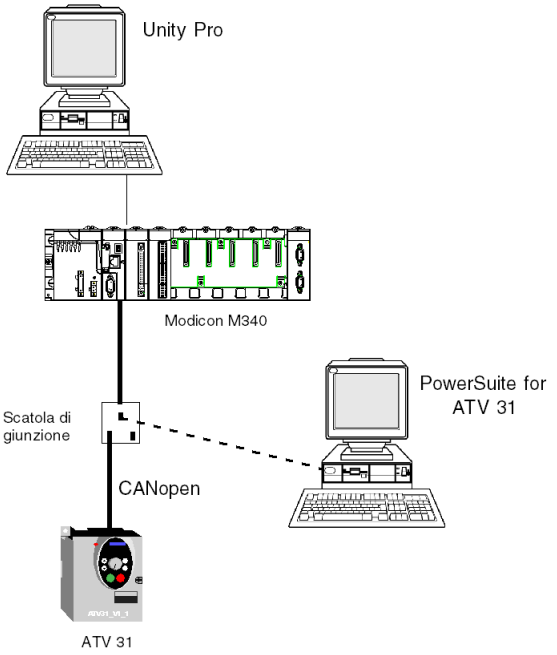
Panoramica

L’architettura descritta è semplice e integra i principi di implementazione del controllo di movimento.

A questo tipo di architettura possono essere aggiunti altri dispositivi per consentire la gestione di più assi.

Illustrazione

Nella seguente figura è illustrata l’architettura utilizzata nell’applicazione che comprende un **ATV 31**.



Requisiti software

Panoramica

In riferimento ai requisiti software indicati nella Guida di avvio rapido *(vedi pagina 13)*, PowerSuite è utilizzato per la configurazione e la regolazione dell'**ATV 31**.

Powersuite for **Lexium 05** consente la regolazione degli assi e offre un metodo semplice per configurare i parametri di un servoazionamento **Lexium 05**.

PowerSuite for **ATV 31** svolge le stesse funzioni, ma per un servoazionamento **ATV 31**.

È possibile configurare alcuni parametri senza PowerSuite tramite l'interfaccia utente *(vedi pagina 147)* del pannello frontale dell'**ATV 31** .

Versioni

Nella seguente tabella sono elencate le versioni hardware e software utilizzate nell'architettura *(vedi pagina 137)* che consentono l'uso degli MFB in Unity Pro.

Hardware	Prima versione del software	Versione del firmware
Modicon M340	Unity Pro V4.0	-
ATV 31	PowerSuite for **ATV 31** V2.00	V1.7 : voce esistente su Unity V3.1 + nuovo profilo MFB per V4.0

NOTA: ATV31 V1.7 è compatibile con le funzioni V1.2.

Requisiti hardware

Codici prodotto dei componenti hardware utilizzati

Nella seguente tabella sono elencati i componenti hardware utilizzati nell'architettura *(vedi pagina 137)* che permettono l'implementazione degli MFB **ATV 31** in Unity Pro.

Hardware	Riferimento
PLC **Modicon M340**	**BMX P34 2030**
Alimentatore **Modicon M340**	**BMX CPS 2000**
Rack **Modicon M340**	**BMX XBP 0800**
Scatola di giunzione CANopen tra **Modicon M340** e il servoazionamento **Lexium 31**	**VW3CANTAP2**
Kit di collegamento al PC	**VW3A8106**
Servoazionamento **ATV 31**	**ATV31H037M2**

NOTA: La resistenza di terminazione è integrata nella scatola di giunzione e deve essere in posizione ON.

11.2 Configurazione del bus CANopen ATV 31

Configurazione dello slave CANopen (ATV 31) sul bus CANopen

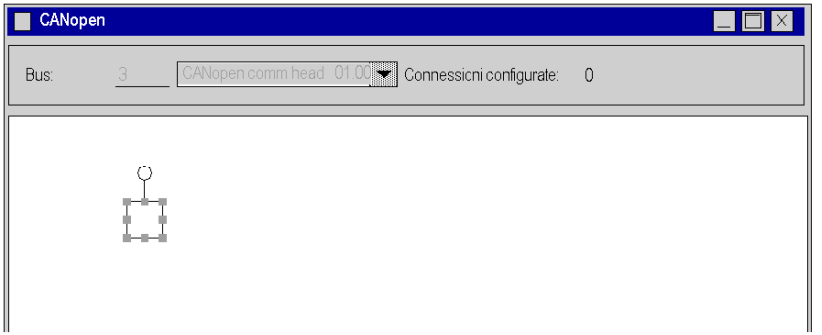
Panoramica

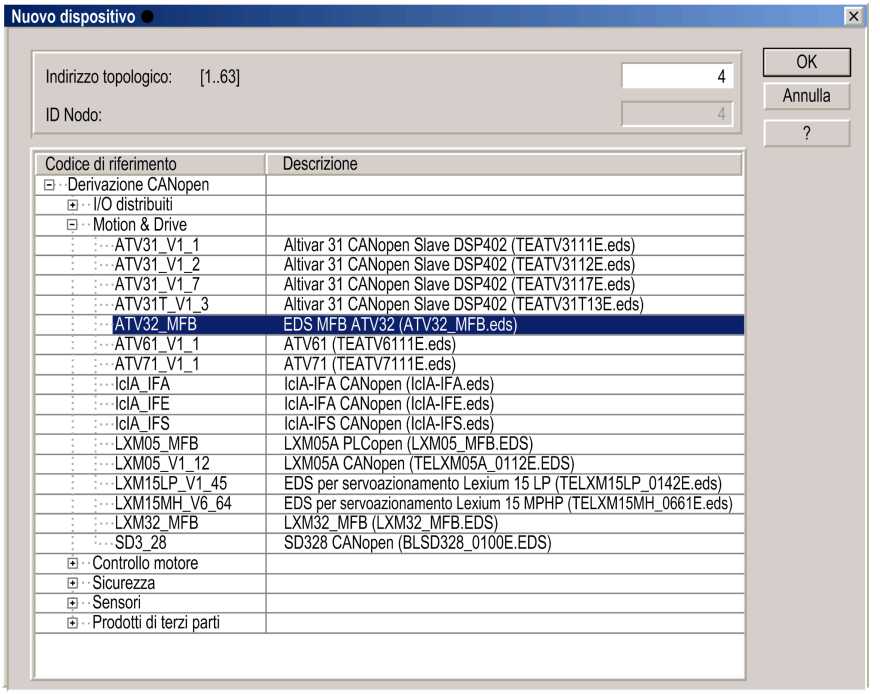
La metodologia per l'implementazione di un bus CANopen tramite Modicon M340 prevede di:

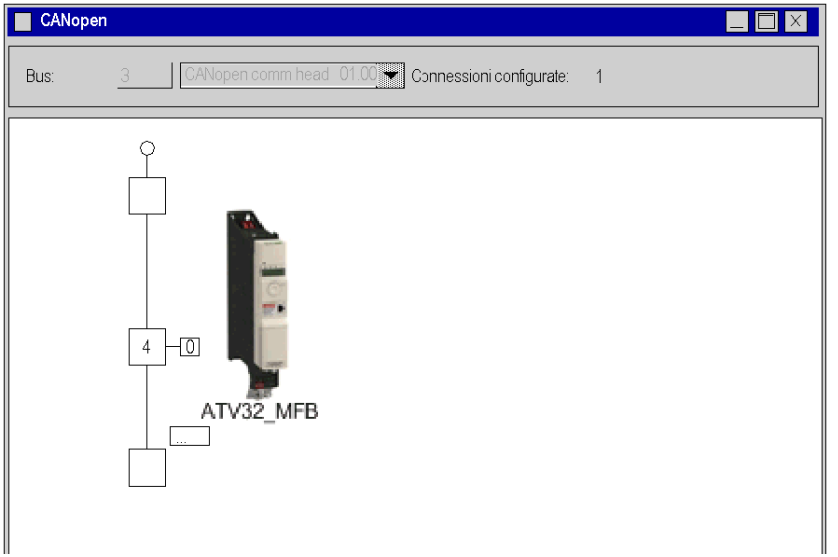
- configurare *(vedi pagina 31)* la porta CANopen per la CPU,
- dichiarare lo slave scelto dal catalogo hardware (vedere paragrafo seguente),
- configurare lo slave,
- abilitare la configurazione mediante Unity Pro,
- verificare *(vedi pagina 35)* il bus CANopen nel browser del progetto.

Configurazione dello slave CANopen

Questa tabella descrive la procedura per configurare lo slave CANopen.

Passo	Azione
1	Nel **Browser del progetto** di Unity Pro, espandere completamente la directory **Configurazione**, quindi fare doppio clic su **CANopen**. **Risultato:** Viene visualizzata la finestra di CANopen: 

Passo	Azione
2	Selezionare**Modifica** →**Nuovo dispositivo**. **Risultato:** viene visualizzata la finestra Nuova apparecchiatura: 
3	Impostare 4 in Indirizzo topologico: Come dispositivo slave scegliere ATV31_V1_2.

Passo	Azione
4	Fare clic su OK per confermare la scelta. **Risultato:** viene visualizzata la finestra CANopen con la nuova apparecchiatura selezionata: 
5	Selezionare**Modifica** →**Apri modulo**. Se MFB non è già selezionato, selezionarlo nell'area Funzione.
6	Il sistema chiede di convalidare le modifiche alla chiusura delle finestre Apparecchiatura e CANopen.

11.3 Configurazione del ATV 31

Scopo di questa sezione

Questa sezione descrive la configurazione di base del servoazionamento usando PowerSuite for **ATV 31** e l'interfaccia utente del pannello frontale del servoazionamento.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Configurazione di ATV 31 in PowerSuite	144
Configurazione di ATV 31 con l'interfaccia utente	147

Configurazione di ATV 31 in PowerSuite

Panoramica

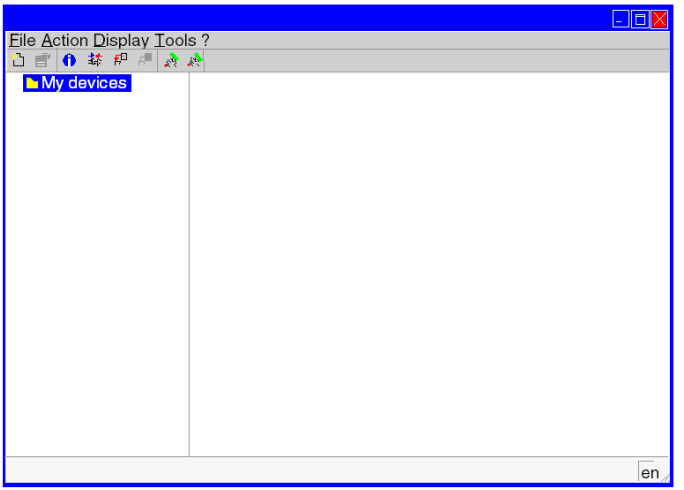
Con PowerSuite, gli utenti possono definire le basi delle apparecchiature installate, descrivere le configurazioni associate e le impostazioni di comunicazione.

PowerSuite offre poi l'accesso a un gruppo di operazioni per la modifica o il trasferimento delle configurazioni nonché per la connessione alle apparecchiature.

Il principio di navigazione di PowerSuite permette di associare un'interfaccia di configurazione con ogni tipo di apparecchiatura rendendone possibile il controllo, la regolazione e il monitoraggio.

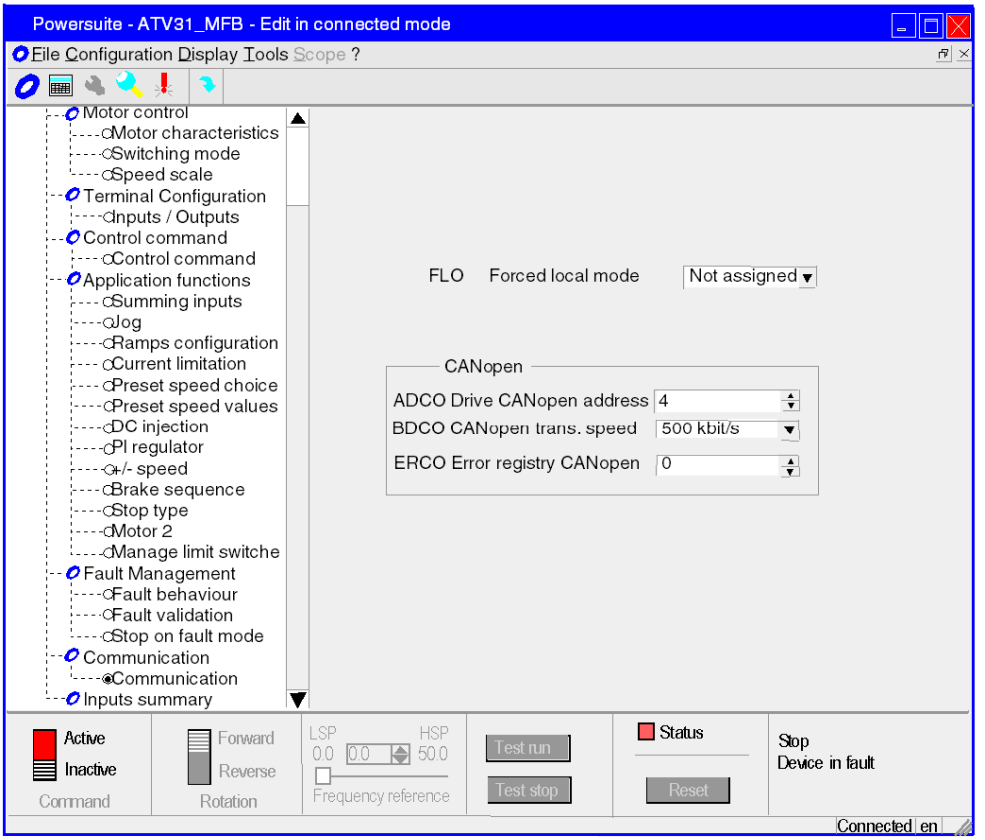
Connessione a ATV 31

Questa tabella descrive la procedura di connessione al servoazionamento **ATV 31**:

Passo	Azione
1	Collegare il PC, su cui è installato PowerSuite per **ATV 31**al connettore **RJ45** del servoazionamento da configurare.
2	Start PowerSuite for **ATV 31** **Risultato:** viene visualizzata la schermata di avvio: 
3	Scegliere **Azione** e poi **Collegamento**. **Risultato:** viene visualizzata una casella di testo.
4	Digitare un nome di progetto (ATV31_MFB) e fare clic su **OK**. **Risultato:** viene visualizzata una finestra di conferma del trasferimento.
5	Premere **Alt F** per avviare il trasferimento dati dal servoazionamento alla stazione di lavoro collegata.

Configurazione di ATV 31

Questa tabella descrive la procedura d'immissione delle impostazioni di base:

Passo	Azione
1	Dopo la connessione e il trasferimento delle configurazioni delle apparecchiature, PowerSuite visualizza una schermata di configurazione in una nuova finestra che offre l'accesso al controllo dell'apparecchiatura e alle funzioni di regolazione e monitoraggio. Uso del comando **Display** →**Configurazione**. Nella struttura ad albero visualizzata, scegliere **Comunicazione** nella directory *Comunicazione*. **Risultato:** viene visualizzata la seguente finestra: 
2	Nella riga **ADCO**, occorre impostare l'indirizzo CANopen su 4.
3	Nella riga **BDCO**, occorre impostare la velocità del bus CANopen su 500.

Passo	Azione
4	Chiudere la finestra per scollegarsi. **Nota:** è possibile regolare le impostazioni del servoazionamento con la stessa procedura. **Risultato:** viene visualizzata la seguente schermata, che mostra i dati salvati in locale:

Configurazione di ATV 31 con l'interfaccia utente

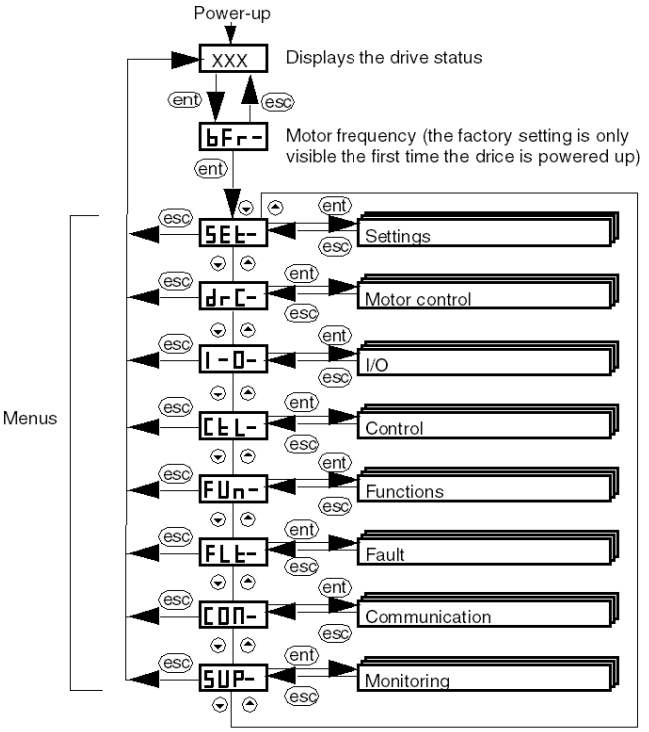
Panoramica

Nel **ATV 31** è integrata un'interfaccia utente. Con questa interfaccia è possibile:

- mettere in linea l'apparecchiatura
- configurararla
- eseguire la diagnostica

Struttura del menu d'interfaccia

Il grafico seguente illustra la procedura d'accesso ai menu d'interfaccia utente:



Impostazioni di base

La seguente tabella descrive la procedura d'immissione delle impostazioni di base (indirizzo e velocità CANopen) con l'interfaccia.

Passo	Azione
1	Premere il pulsante **ENT** sull'interfaccia. **Risultato:** il menu **SET** (Setting) viene visualizzato sull'indicatore di stato dell'interfaccia.
2	Premere il pulsante più volte per accedere al menu **COM**. **Risultato:** il menu **COM** (Communication) viene visualizzato sull'indicatore di stato dell'interfaccia.
3	Premere il pulsante **ENT** sull'interfaccia. **Risultato:** il sottomenu **COAD** (CANopen Address) viene visualizzato sull'indicatore di stato dell'interfaccia.
4	Premere di nuovo **ENT**. **Risultato:** viene visualizzato un valore corrispondente all'indirizzo CANopen dell'apparecchiatura.
5	Premere il pulsante per diminuire il valore o il pulsante per aumentare il valore dell'indirizzo CANopen. Premere **ENT** quando l'indirizzo CANopen desiderato è visualizzato (4). **Risultato:** il valore viene confermato e viene visualizzato di nuovo il sottomenu **COAD** (indirizzo CANopen).
6	Premere il pulsante per accedere al sottomenu **COBD** (velocità in baud di CANopen). Premere **ENT**. **Risultato:** viene visualizzato un valore corrispondente alla velocità CANopen dell'apparecchiatura.
7	Premere il pulsante per aumentare il valore o il pulsante per diminuire il valore della velocità CANopen. Premere **ENT** quando viene visualizzata la velocità CANopen desiderata (500). **Risultato:** il valore viene confermato e viene visualizzato di nuovo il sottomenu **COBD** (velocità in baud CANopen).
8	Premere più volte **ESC** per ritornare al menu principale (**RDY** predefinito).

11.4 Regolazione dell'ATV 31

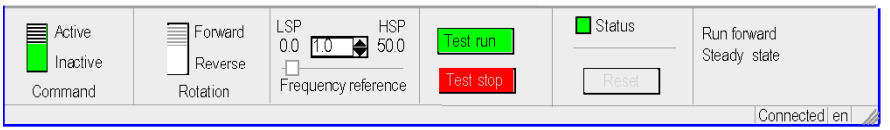
Regolazione dell'ATV 31 con PowerSuite

Preliminarmente

Si raccomanda di regolare la cinematica dell'asse prima che il programma lo avvii automaticamente.

Esempio di regolazione

La seguente tabella offre un esempio di regolazione della cinematica:

Passo	Azione
1	Collegarsi *(vedi pagina 144)* all'**ATV 31**.
2	Dopo la connessione e il trasferimento delle configurazioni delle apparecchiature, PowerSuite apre una nuova finestra con la schermata di configurazione che offre l'accesso al controllo dell'apparecchiatura e alle funzioni di regolazione e monitoraggio. La seguente figura mostra parte della nuova finestra. Questa finestra più in basso permette d'accedere alle funzioni di comando dell'**ATV 31**:
3	Posizionare il cursore della zona **Comando** su **Attivo**.
4	Fare clic sul pulsante **Reset** per azzerare eventuali problemi (se lo stato è rosso).
5	Immettere il valore 1 nella zona **Riferimento frequenza**.
6	Fare clic sul pulsante **Test Run**. **Risultato:** il motore funziona e la sottofinestra è animata: 
7	Posizionare il cursore della zona **Comando** su **Inattivo** una volta che la regolazione è terminata.

ATV 32 - Implementazione dei Blocchi funzione di movimento (Motion)

12

Scopo di questo capitolo

Questa capitolo descrive l'implementazione di un servoazionamento ATV 32 conformemente alla metodologia *(vedi pagina 19)* descritta nella Guida di avvio rapido *(vedi pagina 13)* con un Lexium 05. Il capitolo specifica in dettaglio solo le differenze e le azioni per un ATV 32.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
12.1	Adattamento dell'applicazione all'ATV 32	152
12.2	Configurazione del bus CANopen ATV 32	156
12.3	Configurazione dell'ATV 32	159

12.1 Adattamento dell’applicazione all’ATV 32

Contenuto della sezione

Questa sezione spiega come adattare l’applicazione al sistema **ATV 32** con un tipo di architettura, e indica i requisiti hardware e software.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Architettura di un’applicazione con un ATV 32	153
Requisiti software	154
Requisiti hardware	155

Architettura di un’applicazione con un ATV 32

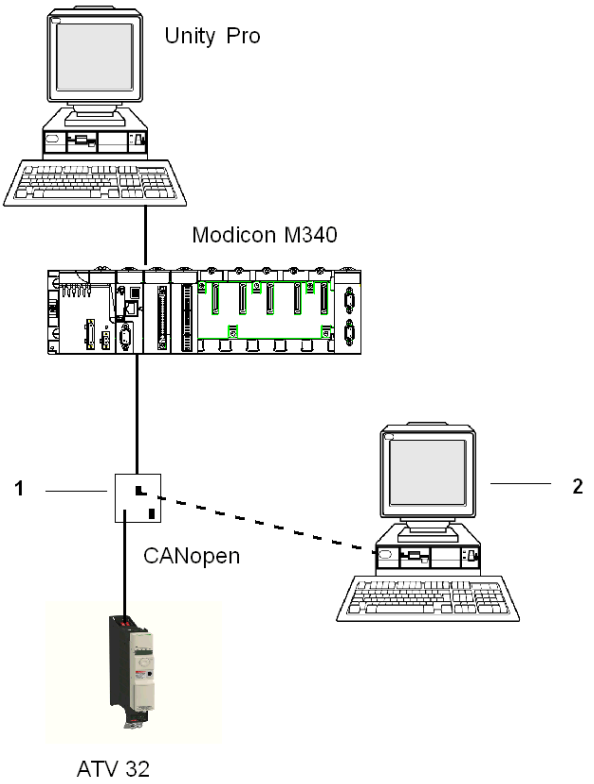
Panoramica

L’architettura descritta è semplice e integra i principi di implementazione del controllo di movimento.

Possono essere aggiunte altre apparecchiature a questa architettura per poter gestire diversi assi.

Illustrazione

Nella seguente figura è illustrata un’architettura usata in un’applicazione che include un **ATV 32**.



1 Scatola di giunzione
2 SoMove

Requisiti software

Panoramica

In riferimento ai requisiti software indicati nella Guida di avvio rapido *(vedi pagina 13)*, SoMove è utilizzato per la configurazione e la regolazione dell'**ATV 32**.

PowerSuite per **Lexium 05** dispone di un metodo semplice per la configurazione dei parametri di un servoazionamento **Lexium 05** servodrive.

Lo stesso metodo di SoMove è altrettanto adeguato per un servoazionamento **ATV 32**.

Se non si utilizza SoMove, è possibile configurare determinati parametri a partire dal pannello frontale **ATV 32**, l'interfaccia utente *(vedi pagina 162)*.

NOTA: Il servoazionamento **ATV 32** non supporta la modalità operativa coppia (torque).

Versioni

Nella seguente tabella sono elencate le versioni hardware e software utilizzate nell'architettura *(vedi pagina 153)* che consentono l'uso degli MFB in Unity Pro:

Hardware	Prima versione del software	Versione del firmware
Modicon M340	Unity Pro V6.0 o successivo	-
ATV 32	SoMove	V1.2

Requisiti hardware

Codici prodotto dei componenti hardware utilizzati

Nella seguente tabella sono elencati i componenti hardware utilizzati nell'architettura *(vedi pagina 153)* che permettono l'implementazione degli MFB **ATV 32** in Unity Pro:

Hardware	**Riferimento**
PLC **Modicon M340**	**BMX P34 2030**
Alimentatore **Modicon M340**	**BMX CPS 2000**
Rack **Modicon M340**	**BMX XBP 0800**
Scatola di giunzione CANopen tra **Modicon M340** e il servoazionamento **Lexium 32**	**VW3CANTAP2**
Kit di collegamento al PC	**VW3A8106**
Servoazionamento **ATV 32**	**ATV32••••••**

NOTA: La resistenza di terminazione è integrata nella scatola di giunzione e deve essere in posizione ON.

12.2 Configurazione del bus CANopen ATV 32

Configurazione dello slave CANopen (ATV 32) sul bus CANopen

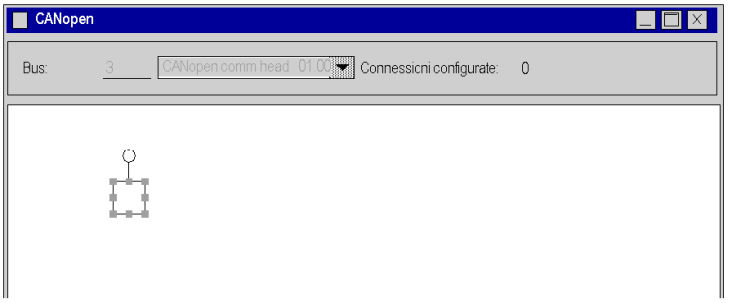
Panoramica

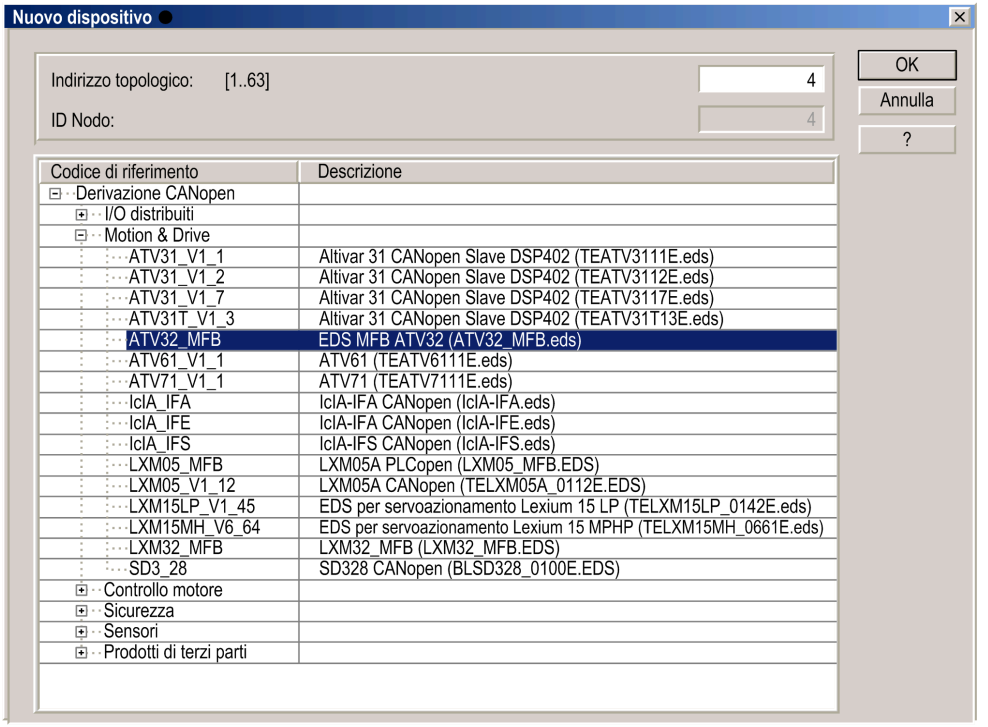
La metodologia per l'implementazione di un bus CANopen tramite Modicon M340 prevede di:

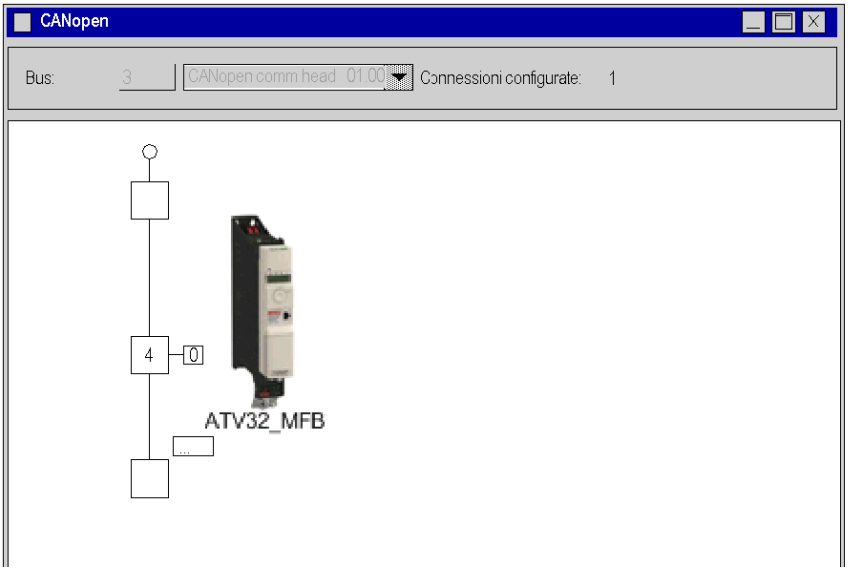
- configurare *(vedi pagina 31)* la porta CANopen della CPU
- dichiarare lo slave scelto dal catalogo hardware (vedere paragrafo seguente)
- configurare lo slave
- attivare la configurazione usando Unity Pro
- verificare *(vedi pagina 35)* il bus CANopen nel browser di progetto

Configurazione dello slave CANopen

Questa tabella descrive la procedura per configurare lo slave CANopen:

Passo	Azione
1	Nel **Browser del progetto** di Unity Pro, espandere completamente la directory **Configurazione**, quindi fare doppio clic su **CANopen**. **Risultato:** viene visualizzata la finestra: 

Passo	Azione
2	Selezionare **Modifica** →**Nuova apparecchiatura**. **Risultato:** viene visualizzata la finestra Nuova apparecchiatura: 
3	Impostare 4 nell'indirizzo topologico. Come dispositivo slave scegliere ATV32_V1_2.

Passo	Azione
4	Fare clic su OK per confermare la scelta. **Risultato:** viene visualizzata la finestra CANopen con la nuova apparecchiatura selezionata: 
5	Selezionare **Modifica** →**Apri modulo**. Se MFB non è già selezionato, selezionarlo nell'area Funzione.
6	Convalidare le modifiche effettuate quando si chiudono le finestre Dispositivi e CANopen.

12.3 Configurazione dell’ATV 32

Contenuto della sezione

Questa sezione descrive le configurazioni di base del servoazionamento usando SoMove e l’interfaccia utente del pannello frontale del servoazionamento.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Configurazione del servoazionamento ATV 32 con SoMove	160
Configurazione dell’ATV 32 con l’interfaccia utente	162

Configurazione del servoazionamento ATV 32 con SoMove

Panoramica

Con SoMove, gli utenti possono definire le basi delle apparecchiature installate, descrivere le configurazioni associate e le impostazioni di comunicazione.

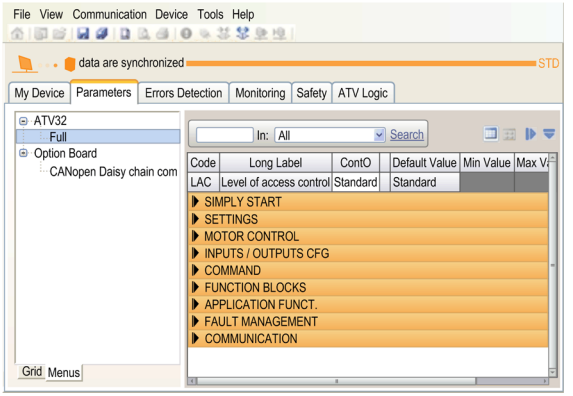
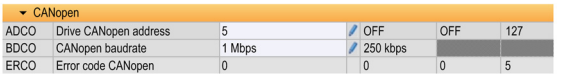
SoMove offre poi l'accesso a un gruppo di operazioni per la modifica o il trasferimento delle configurazioni nonché per la connessione alle apparecchiature.

Il principio di navigazione di SoMove permette di associare un'interfaccia di configurazione con ogni tipo di apparecchiatura rendendone possibile il controllo, la regolazione e il monitoraggio.

Connessione dell'ATV 32

Questa tabella descrive la procedura di connessione al servoazionamento **ATV 32**:

Passo	Operazione
1	Collegare il PC, che dispone di SoMove per l'ATV 32 installato, al connettore **RJ45** sul servoazionamento da configurare.
2	Avviare SoMove. **Risultato:** viene visualizzata la schermata di avvio:

Passo	Operazione
3	Scegliere **Connetti**. **Risultato:**viene visualizzata la seguente schermata:
4	Per confermare l'uso di queste istruzioni, premere **Alt+F**. **Risultato:**viene visualizzata la seguente schermata: 
5	Aprire la scheda **Comunicazione e la scheda →CANopen** **Risultato:**viene visualizzata la seguente schermata: 
6	Alla riga ADCO, impostare l'indirizzo CANopen su 4
7	Alla riga BCDO, impostare la velocità di trasmissione CANopen a 500 kbps.
8	Scollegare la workstation dal servoazionamento.
9	Salvare il progetto utilizzando **ATV32_MFB** come il nome del progetto.

Configurazione dell'ATV 32 con l'interfaccia utente

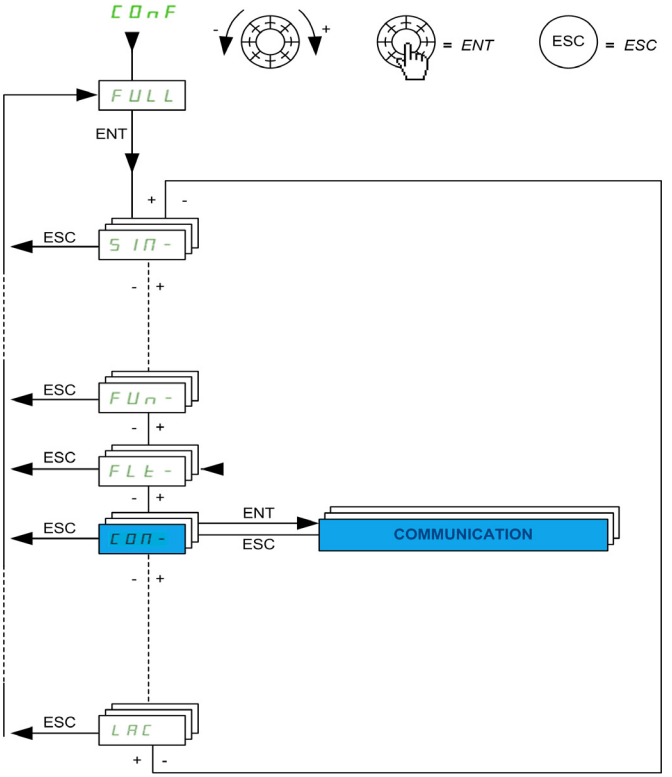
Panoramica

Il servoazionamento **ATV 32** integra un'interfaccia utente. Con questa interfaccia è possibile:

- mettere il dispositivo online
- configurare il dispositivo
- regolare le impostazioni
- eseguire la diagnostica

Struttura del menu d'interfaccia

Il grafico seguente mostra come accedere ai menu di Configurazione utilizzando Jog Dial per accedere al menu `COnF`:



ENT Invio
ESC Escape

Impostazioni di base CANopen

Questa tabella contiene la procedura d'immissione delle impostazioni di base per la velocità e l'indirizzo CANopen attraverso l'interfaccia utente:

Passo	Operazione
1	Usare Jog dial per selezionare **COnF**. **Risultato:** Viene visualizzato il menu **COnF** (configurazione CANopen).
2	Premere il tasto **ENT**. **Risultato:** viene visualizzato un elenco a scorrimento di sotto menu.
3	Usare Jog dial per selezionare **FULL**. **Risultato**: Viene visualizzato il menu **FULL** (parametri non precaricati).
4	Premere il tasto **ENT**. **Risultato**: viene visualizzato un elenco a scorrimento di sotto menu.
5	Usare Jog dial per selezionare **COM**. **Risultato**: Viene visualizzato il menu **COM** (Comunicazione).
6	Premere il tasto **ENT**. **Risultato**: viene visualizzato un elenco a scorrimento di sotto menu.
7	Usare Jog dial per selezionare **CnO**. **Risultato**: Viene visualizzato il menu **CnO** (CANopen).
8	Premere il tasto **ENT**. **Risultato**: Viene visualizzato un elenco di parametri
9	Usare Jog dial per selezionare **AdCO**. **Risultato**: Viene visualizzato il parametro **AdCO** (indirizzo CANopen).
10	Premere il tasto **ENT**. **Risultato**: Viene visualizzato un valore corrispondente all'indirizzo CANopen predefinito.
11	Utilizzare Jog dial per scegliere l'indirizzo CANopen (4). **Risultato**: Viene visualizzato l'indirizzo CANopen selezionato.
12	Premere il tasto **ENT**. **Risultato**: Viene visualizzato il parametro **AdCO** (indirizzo CANopen).
13	Usare Jog dial per selezionare **bdCO**. **Risultato**: Viene visualizzato il parametro **bdCO** (velocità CANopen)
14	Premere il tasto **ENT**. **Risultato**: Viene visualizzato un valore corrispondente alla velocità CANopen predefinita.
15	Utilizzare Jog dial per scegliere la velocità CANopen (500). **Risultato**: Viene visualizzata la velocità CANopen selezionata.
16	Premere il tasto **ENT**. **Risultato**: Viene visualizzato il parametro **bdCO** (velocità CANopen).
17	Premere **ESC** diverse volte per tornare al menu principale.

ATV 71 Implementazione dei Blocchi funzione di movimento

13

Scopo di questo capitolo

Questa capitolo presenta l'implementazione di un servoazionamento ATV 71 conformemente alla metodologia *(vedi pagina 19)* descritta nella guida di avvio rapido *(vedi pagina 13)* con un Lexium 05. Il capitolo specifica in dettaglio solo le differenze e le azioni per un ATV 71.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
13.1	Adattamento dell'applicazione all'ATV 71	166
13.2	Configurazione del bus CANopen ATV 71	170
13.3	Configurazione dell'ATV 71	173
13.4	Regolazione dell'ATV 71	179

13.1 Adattamento dell’applicazione all’ATV 71

Scopo di questa sezione

In questa sezione sono descritte le operazioni da eseguire per adattare l’applicazione all’**ATV 71** con un’architettura e sono indicati i requisiti hardware e software.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Architettura di un’applicazione con un ATV 71	167
Requisiti software	168
Requisiti hardware	169

Architettura di un’applicazione con un ATV 71

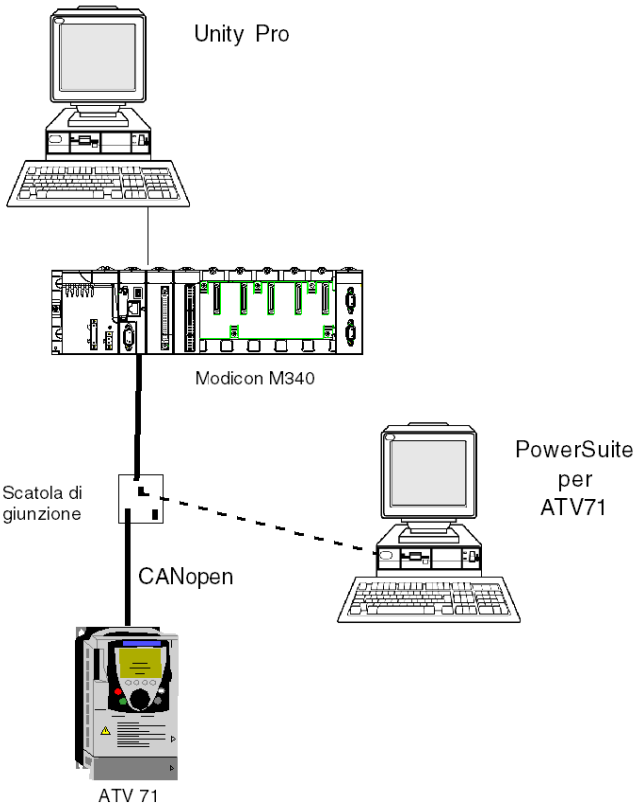
Panoramica

L’architettura descritta è semplice e integra i principi di implementazione del controllo di movimento.

A questo tipo di architettura possono essere aggiunti altri dispositivi per consentire la gestione di più assi.

Illustrazione

Nella seguente figura è illustrata l’architettura utilizzata nell’applicazione che comprende un **ATV 71**.



Requisiti software

Panoramica

In riferimento ai requisiti software indicati nella Guida di avvio rapido *(vedi pagina 13)*, PowerSuite è utilizzato per la configurazione e la regolazione dell'**ATV 71**.

PowerSuite for **Lexium 05** consente la regolazione degli assi e offre un metodo semplice per configurare i parametri di un servoazionamento **Lexium 05**.

PowerSuite for **ATV 71** svolge le stesse funzioni, ma per un servoazionamento **ATV 71**.

È possibile configurare alcuni parametri senza PowerSuite utilizzando l' interfaccia utente *(vedi pagina 177)* del pannello frontale dell'**ATV 71**.

Versioni

Nella seguente tabella sono elencate le versioni hardware e software utilizzate nell'architettura *(vedi pagina 167)* che consentono l'uso degli MFB in Unity Pro.

Hardware	Prima versione del software	Versione del firmware
Modicon M340	Unity Pro V4.0	-
ATV 71	PowerSuite for **ATV 71** V2.00	Compatibile da V1.1, V 1.7 gestita da MTM

Requisiti hardware

Codici prodotto dei componenti hardware utilizzati

Nella seguente tabella sono elencati i componenti hardware utilizzati nell'architettura *(vedi pagina 167)* che permettono l'implementazione degli MFB **ATV 71** in Unity Pro.

Hardware	Riferimento
PLC **Modicon M340**	**BMX P34 2030**
Alimentatore **Modicon M340**	**BMX CPS 2000**
Rack **Modicon M340**	**BMX XBP 0800**
Scatola di giunzione CANopen tra **Modicon M340** e il servoazionamento **Lexium 71**	**VW3CANTAP2**
Cavo di programmazione RJ45 con adattatore RS485/RS232 tra la scatola di giunzione e il servoazionamento	**ACC2CRAAEF030**
Servoazionamento **ATV 71**	**ATV71H075N2Z**

NOTA: La resistenza di terminazione è integrata nella scatola di giunzione e deve essere in posizione ON.

13.2 Configurazione del bus CANopen ATV 71

Configurazione dello slave CANopen (ATV 71) sul bus CANopen

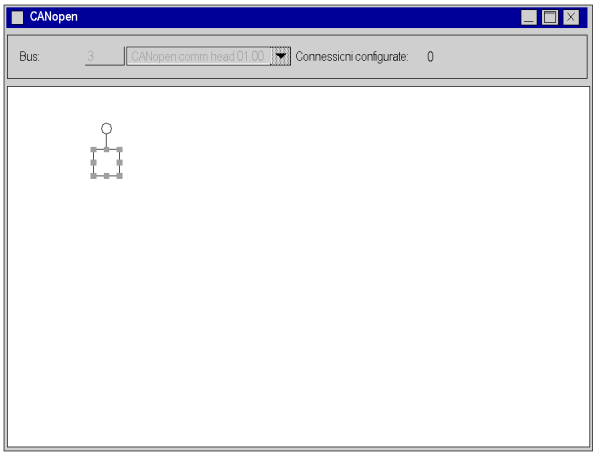
Panoramica

La metodologia per l'implementazione di un bus CANopen tramite Modicon M340 prevede di:

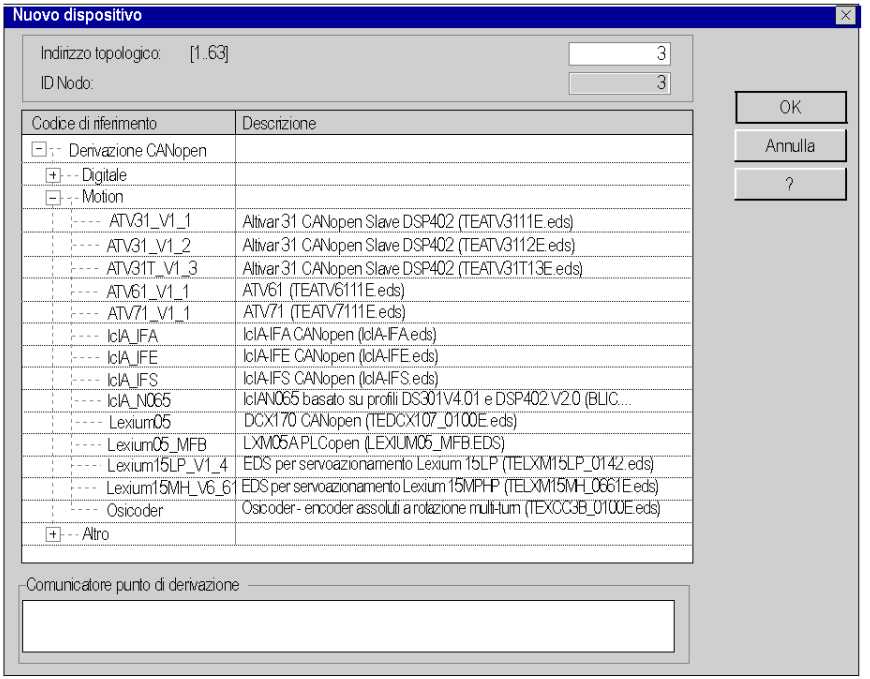
- configurare *(vedi pagina 31)* la porta CANopen per la CPU,
- dichiarare lo slave scelto dal catalogo hardware (vedere paragrafo seguente),
- configurare lo slave,
- abilitare la configurazione mediante Unity Pro,
- verificare *(vedi pagina 35)* il bus CANopen nel browser del progetto.

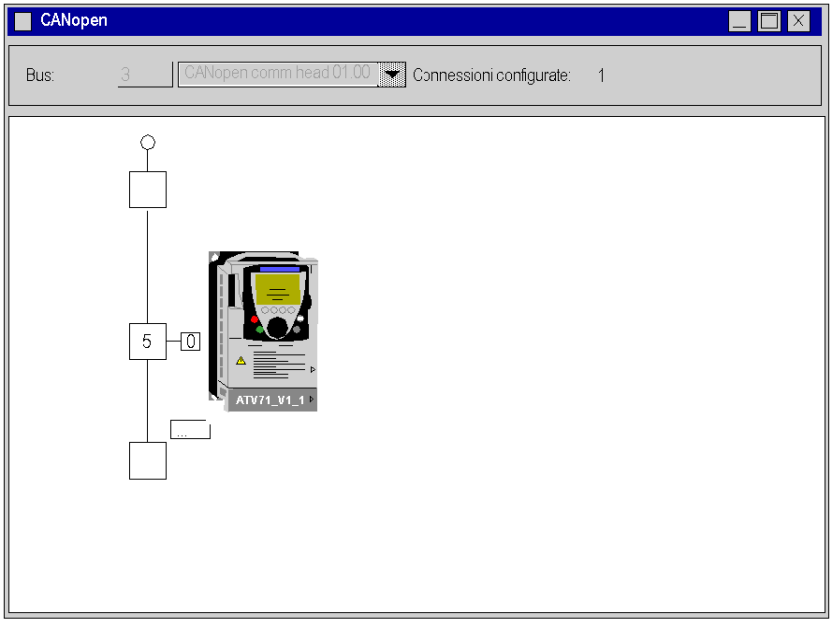
Configurazione dello slave CANopen

Questa tabella descrive la procedura per configurare lo slave CANopen.

Passo	Azione
1	Nel **Browser del progetto** di Unity Pro, espandere completamente la directory **Configurazione**, quindi fare doppio clic su **CANopen**. **Risultato:** viene visualizzata la finestra CANopen: 

Passo	Azione
2	Selezionare **Modifica** →**Nuova apparecchiatura**. **Risultato:** viene visualizzata la finestra Nuova apparecchiatura:
3	Impostare 5 in Indirizzo topologico: Come dispositivo slave scegliere ATV71_V1_1.



Passo	Azione
4	Fare clic su OK per confermare la scelta. **Risultato:** viene visualizzata la finestra CANopen con la nuova apparecchiatura selezionata: 
5	Selezionare **Modifica →Apri modulo**. Se MFB non è già selezionato, selezionarlo nell'area Funzione.
6	Il sistema chiede di convalidare le modifiche alla chiusura delle finestre Apparecchiatura e CANopen.

13.3 Configurazione dell’ATV 71

Scopo di questa sezione

Questa sezione descrive la configurazione di base del servoazionamento usando PowerSuite per **ATV 71** e l’interfaccia utente del pannello frontale del servoazionamento.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Configurazione di ATV 71 in PowerSuite	174
Configurazione di ATV 71 con l’interfaccia utente	177

Configurazione di ATV 71 in PowerSuite

Panoramica

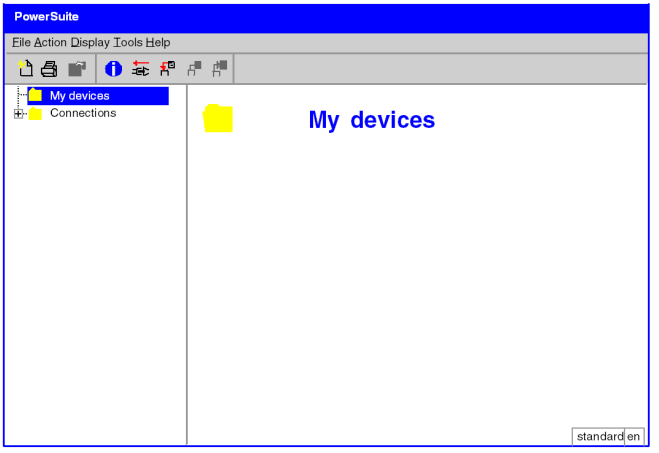
Con PowerSuite, gli utenti possono definire le basi delle apparecchiature installate, descrivere le configurazioni associate e le impostazioni di comunicazione.

PowerSuite offre poi l'accesso a un gruppo di operazioni per la modifica o il trasferimento delle configurazioni nonché per la connessione alle apparecchiature.

Il principio di navigazione di PowerSuite permette di associare un'interfaccia di configurazione con ogni tipo di apparecchiatura rendendone possibile il controllo, la regolazione e il monitoraggio.

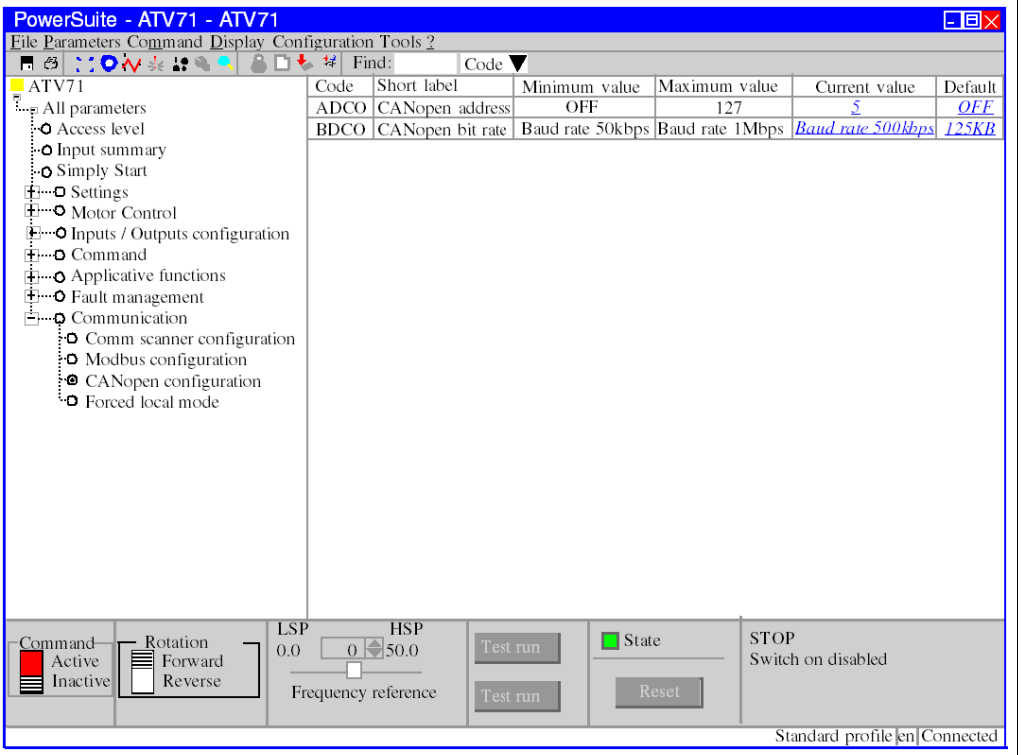
Connessione a ATV 71

Questa tabella descrive la procedura di connessione al servoazionamento **ATV 71**:

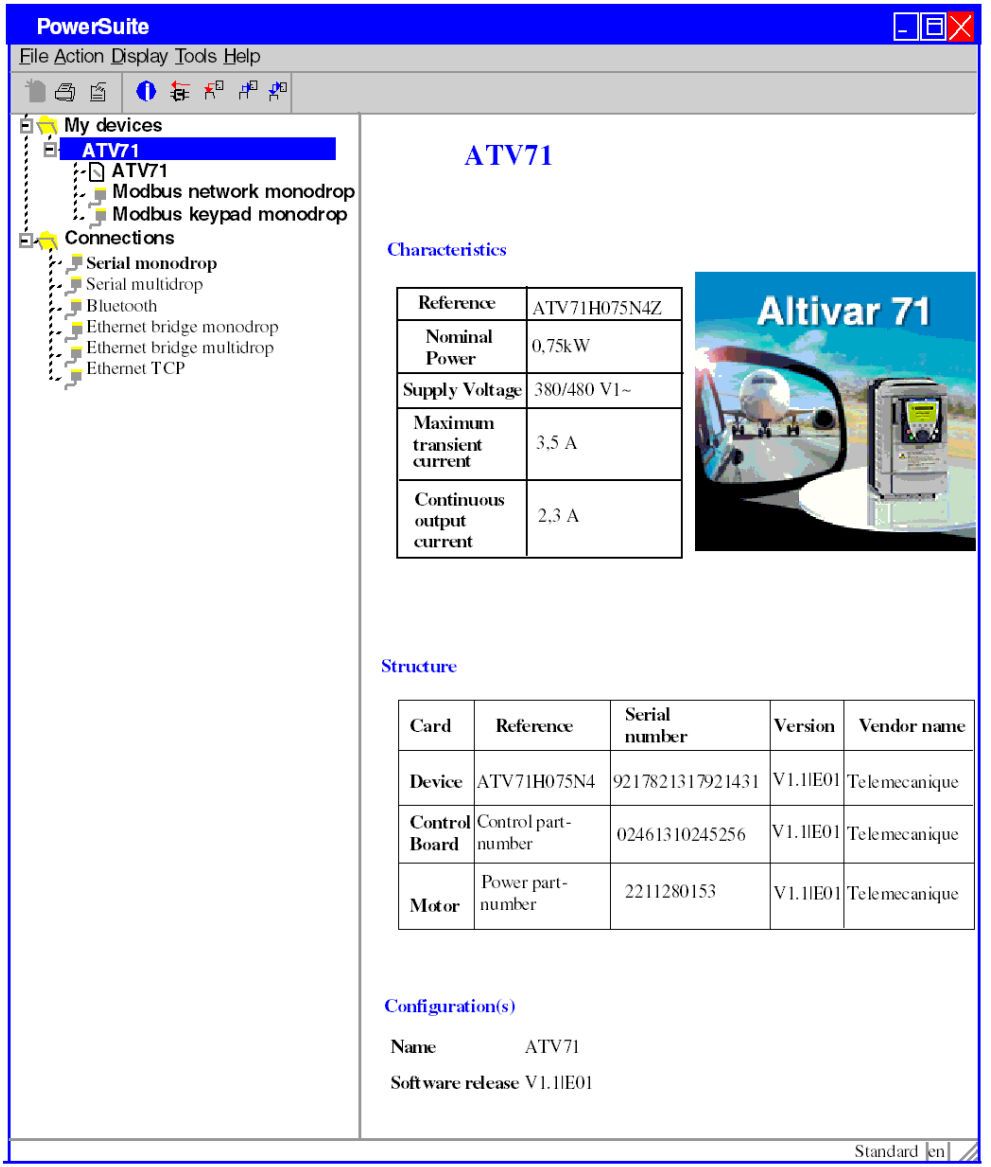
Passo	Azione
1	Collegare il PC, su cui è installato PowerSuite per **ATV 71**al connettore **RJ45** del servoazionamento da configurare.
2	Start PowerSuite for **ATV 71** **Risultato:** viene visualizzata la schermata di avvio: 
3	Scegliere **Azione** e poi **Collegamento**. **Risultato:** viene visualizzata una casella di testo.
4	Digitare un nome di progetto (ATV71_MFB) e fare clic su **OK**. **Risultato:** viene visualizzata una finestra di conferma del trasferimento.
5	Premere **Alt F** per avviare il trasferimento dati dal servoazionamento alla stazione di lavoro collegata.

Configurazione di base dell'ATV 71

Questa tabella descrive la procedura d'immissione delle impostazioni di base:

Passo	Azione
1	Dopo la connessione e il trasferimento delle configurazioni delle apparecchiature, PowerSuite visualizza una schermata di configurazione in una nuova finestra che offre l'accesso al controllo dell'apparecchiatura e alle funzioni di regolazione e monitoraggio. Nella struttura ad albero visualizzata, scegliere **Comunicazione** nella directory *Comunicazione*. **Risultato:** viene visualizzata la seguente finestra: 
2	Nella riga **ADCO**, occorre impostare l'indirizzo CANopen su 5.
3	Nella riga **BDCO**, occorre impostare la velocità del bus CANopen su 500. **Nota:** è possibile regolare le impostazioni del servoazionamento con la stessa procedura.

Passo	Azione
4	Una volta regolate le impostazioni, usare il comando **Configurazione** →**Scollega** per scollegarsi. **Risultato:** viene visualizzata la seguente schermata, che mostra i dati salvati in locale:



Configurazione di ATV 71 con l'interfaccia utente

Panoramica

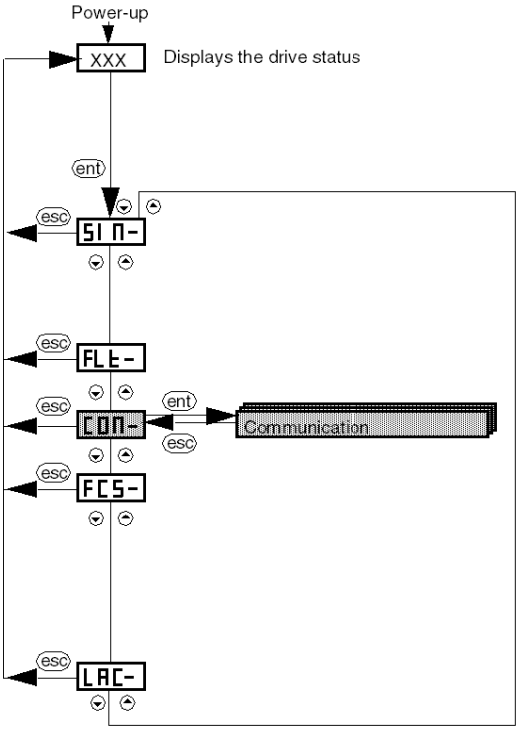
Nel **ATV 71** è integrata un'interfaccia utente. Con questa interfaccia è possibile:

- mettere in linea l'apparecchiatura
- configurararla
- eseguire la diagnostica

NOTA: esiste un terminale grafico più facile da usare, ad esempio nella diagnsotica degli errori.

Struttura del menu d'interfaccia

Il grafico seguente illustra la procedura d'accesso ai menu d'interfaccia utente:



Impostazioni di base

La seguente tabella descrive la procedura d'immissione delle impostazioni di base (indirizzo e velocità CANopen) con l'interfaccia.

Passo	Azione
1	Premere il pulsante **ENT** sull'interfaccia. **Risultato:** il menu **SET** (Setting) viene visualizzato sull'indicatore di stato dell'interfaccia.
2	Premere il pulsante più volte per accedere al menu **COM**. **Risultato:** il menu **COM** (Communication) viene visualizzato sull'indicatore di stato dell'interfaccia.
3	Premere il pulsante **ENT** sull'interfaccia. **Risultato:** il sottomenu **COAD** (CANopen Address) viene visualizzato sull'indicatore di stato dell'interfaccia.
4	Premere di nuovo **ENT**. **Risultato:** viene visualizzato un valore corrispondente all'indirizzo CANopen dell'apparecchiatura.
5	Premere il pulsante per diminuire il valore o il pulsante per aumentare il valore dell'indirizzo CANopen. Premere **ENT** quando l'indirizzo CANopen desiderato è visualizzato (5). **Risultato:** il valore viene confermato e viene visualizzato di nuovo il sottomenu **COAD** (indirizzo CANopen).
6	Premere il pulsante per accedere al sottomenu **COBD** (velocità in baud di CANopen). Premere **ENT**. **Risultato:** viene visualizzato un valore corrispondente alla velocità CANopen dell'apparecchiatura.
7	Premere il pulsante per aumentare il valore o il pulsante per diminuire il valore della velocità CANopen. Premere **ENT** quando viene visualizzata la velocità CANopen desiderata (500). **Risultato:** il valore viene confermato e viene visualizzato di nuovo il sottomenu **COBD** (velocità in baud CANopen).
8	Premere più volte **ESC** per ritornare al menu principale (**RDY** predefinito).

13.4 Regolazione dell'ATV 71

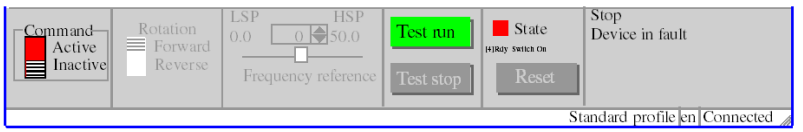
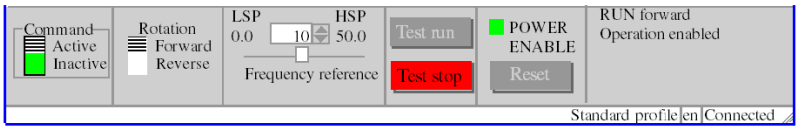
Regolazione dell'ATV 71 con PowerSuite

Preliminarmente

Si raccomanda di regolare la cinematica dell'asse prima che il programma lo avvii automaticamente.

Esempio di regolazione

La seguente tabella offre un esempio di regolazione della cinematica:

Passo	Azione
1	Collegarsi *(vedi pagina 174)* all'**ATV 71**.
2	Dopo la connessione e il trasferimento delle configurazioni delle apparecchiature, PowerSuite apre una nuova finestra con la schermata di configurazione che offre l'accesso al controllo dell'apparecchiatura e alle funzioni di regolazione e monitoraggio. La seguente figura mostra parte della nuova finestra. Questa finestra più in basso permette d'accedere alle funzioni di comando dell'**ATV 71**: 
3	Posizionare il cursore della zona **Comando** su **Attivo**.
4	Fare clic sul pulsante **Reset** per azzerare tutti i problemi.
5	Immettere il valore 10 nella zona **Riferimento frequenza**.
6	Fare clic sul pulsante **Test Run**. **Risultato:** il motore funziona e la sottofinestra è animata: 
7	Posizionare il cursore della zona **Comando** su **Inattivo** una volta che la regolazione è terminata.

IclA Implementazione dei Blocchi funzione di movimento

14

Scopo di questo capitolo

Questa capitolo presenta l'implementazione di un servoazionamento IclA conformemente alla metodologia *(vedi pagina 19)* descritta nella guida di avvio rapido *(vedi pagina 13)* con un Lexium 05. Il capitolo specifica in dettaglio solo le differenze e le azioni per un IclA.

Contenuto di questo capitolo

Questo capitolo contiene le seguenti sezioni:

Sezione	Argomento	Pagina
14.1	Adattamento dell'applicazione all'IclA	182
14.2	Configurazione del bus CANopen IclA	186
14.3	Configurazione dell'IclA	189
14.4	Regolazione dell'IclA	190

14.1 Adattamento dell’applicazione all’IclA

Scopo di questa sezione

In questa sezione sono descritte le operazioni da eseguire per adattare l’applicazione all’**IclA** con un’architettura e sono indicati i requisiti hardware e software.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Architettura di un’applicazione con un IclA	183
Requisiti software	184
Requisiti hardware	185

Architettura di un’applicazione con un IclA

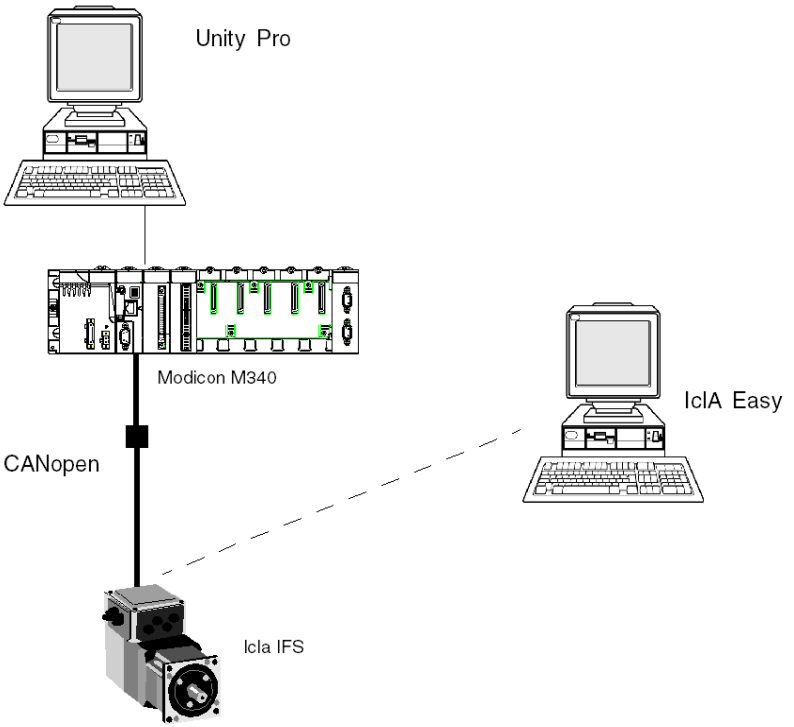
Panoramica

L’architettura descritta è semplice e integra i principi di implementazione del controllo di movimento.

A questo tipo di architettura possono essere aggiunti altri dispositivi per consentire la gestione di più assi.

Illustrazione

Nella seguente figura è illustrata l’architettura utilizzata nell’applicazione che comprende un **IclA IFS**.



Requisiti software

Panoramica

In riferimento ai requisiti software indicati nella guida di avvio rapido *(vedi pagina 13)*, IclA Easy è utilizzato per la configurazione e la regolazione dell'**IclA**.

PowerSuite for **Lexium 05** consente la regolazione degli assi e offre un metodo semplice per configurare i parametri di un servoazionamento **Lexium 05**.

IclA Easy svolge le stesse funzioni per un servoazionamento **IclA**.

È necessario configurare alcuni parametri senza Icla Easy utilizzando gli switch *(vedi pagina 189)***IclA**, essendo questo l'unico modo per configurare tali parametri.

Versioni

Nella seguente tabella sono elencate le versioni hardware e software utilizzate nell'architettura *(vedi pagina 183)* che consentono l'uso degli MFB in Unity Pro.

Hardware	Prima versione del software	Versione del firmware
Modicon M340	Unity Pro V4.0	-
IclA	**EasyIclA** V1.104	**IclA IFA** compatibile a partire da V1.1007 **IclA IFE** compatibile a partire da V1.1007 **IclA IFS** compatibile a partire da V1.1007

Requisiti hardware

Codici prodotto dei componenti hardware utilizzati

Nella seguente tabella sono elencati i componenti hardware utilizzati nell'architettura *(vedi pagina 183)* che permettono l'implementazione degli MFB **IclA** in Unity Pro.

Hardware	Riferimento
PLC **Modicon M340**	**BMX P34 2030**
Alimentatore **Modicon M340**	**BMX CPS 2000**
Rack **Modicon M340**	**BMX XBP 0800**
Il connettore femmina CANopen SUB-D9-Way (piegato a 90° + connettore SUB-D9-Way per collegare un PC sul bus)	**TSX CAN KCDF 90TP**
Gruppo di cavi preassemblati CANopen con connettori femmina prestampati SUB-D9-Way a entrambi i capi	**TSX CAN CADD03**
Dongle PCAN PS/2 per IclA Easy (convertitore da parallela a CAN)	**IPEH-002019**
Cavo CANopen	**TSX CAN CA50**
Servoazionamento **IclA**	**IFS61/2-CAN-DS/-I-B54/0-001RPP41**

NOTA: La resistenza di terminazione è integrata nell'**IclA** e deve essere sulla posizione ON *(vedi pagina 189)*.

14.2 Configurazione del bus CANopen IclA

Configurazione dello slave CANopen (IclA) sul bus CANopen

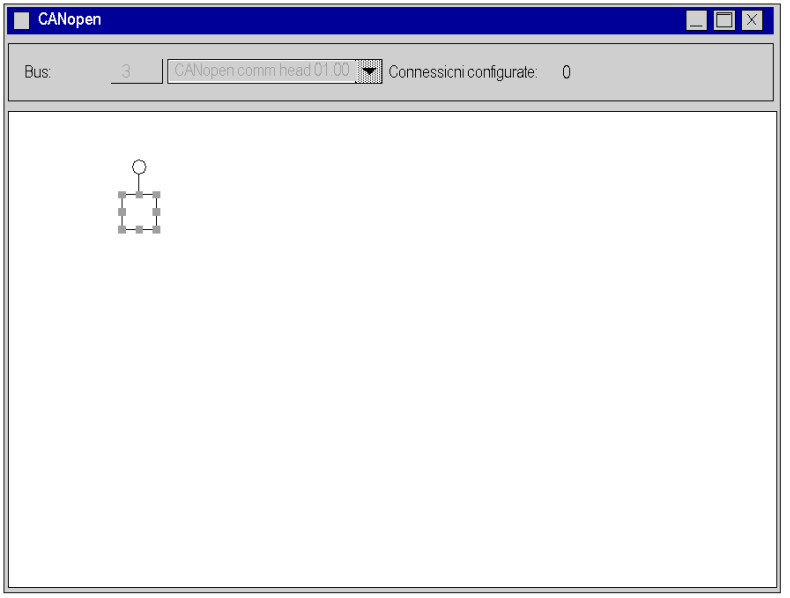
Panoramica

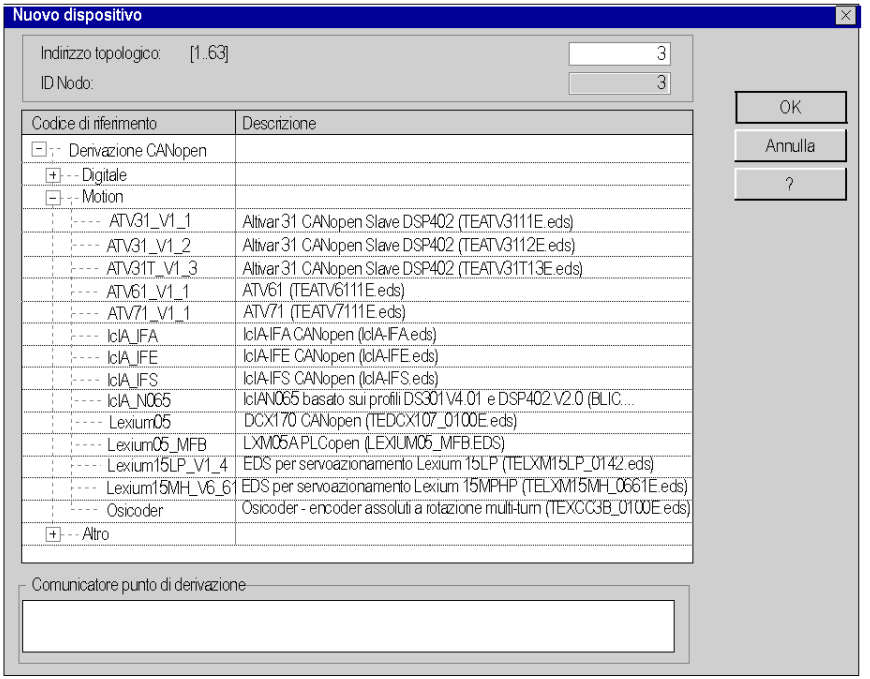
La metodologia per l'implementazione di un bus CANopen tramite Modicon M340 prevede di:

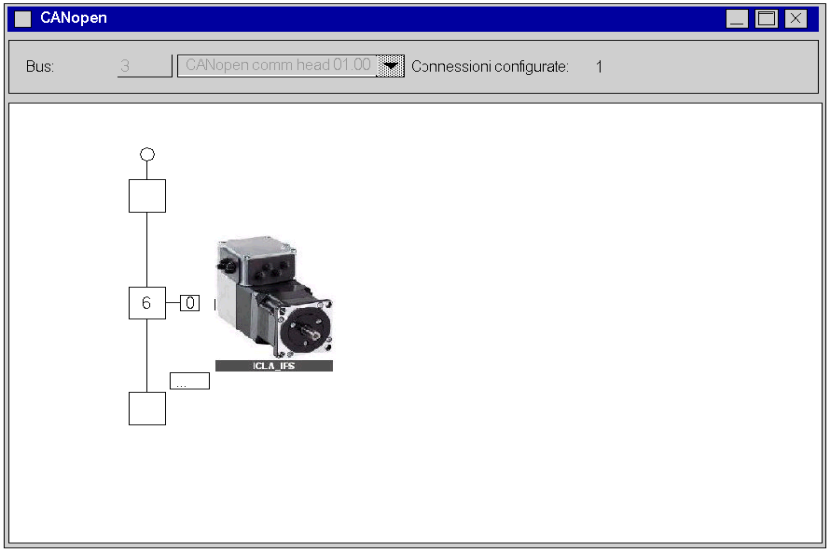
- configurare *(vedi pagina 31)* la porta CANopen per la CPU,
- dichiarare lo slave scelto dal catalogo hardware (vedere paragrafo seguente),
- configurare lo slave,
- abilitare la configurazione mediante Unity Pro,
- verificare *(vedi pagina 35)* il bus CANopen nel browser del progetto.

Configurazione dello slave CANopen

Questa tabella descrive la procedura per configurare lo slave CANopen.

Passo	Azione
1	Nel **Browser del progetto** di Unity Pro, espandere completamente la directory **Configurazione**, quindi fare doppio clic su **CANopen**. **Risultato:** viene visualizzata la finestra CANopen: 

Passo	Azione
2	Selezionare **Modifica** →**Nuova apparecchiatura**. **Risultato:** viene visualizzata la finestra Nuova apparecchiatura: 
3	Impostare 6 in Indirizzo topologico: Come dispositivo slave scegliere IcIA_IFS.

Passo	Azione
4	Fare clic su OK per confermare la scelta. **Risultato:** viene visualizzata la finestra CANopen con la nuova apparecchiatura selezionata: 
5	Selezionare **Modifica** →**Apri modulo**. Se MFB non è già selezionato, selezionarlo nell'area Funzione.
6	Il sistema chiede di convalidare le modifiche alla chiusura delle finestre Apparecchiatura e CANopen.

14.3 Configurazione dell'IclA

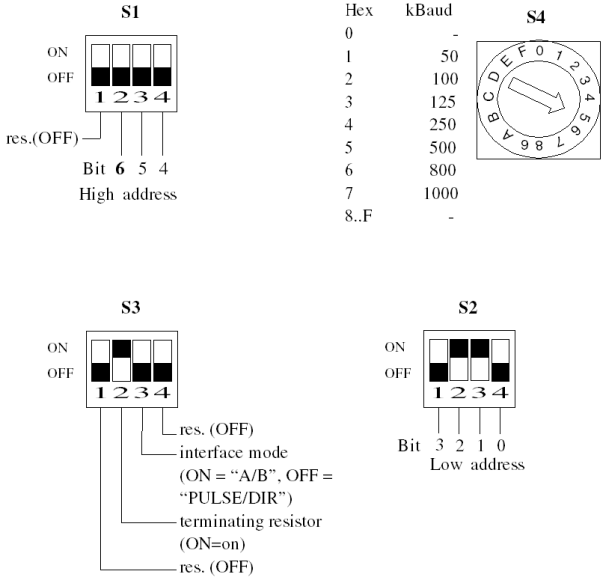
Configurazione dell'IclA con i microinterruttori DIP

Panoramica

L'indirizzo e la velocità di trasmissione vengono impostati con i microinterruttori DIP posti sull'azionamento**IclA IFX**.

Microinterruttori DIP

La seguente illustra mostra gli interruttori DIP all'interno dell'azionamento:



Impostazioni di base

La velocità di trasmissione è impostata con il microinterruttore S4 sulla posizione 5 per un valore di 500.

L'indirizzo CANopen è impostato con i microinterruttori S1 e S2. Impostare S2.3 e S2.2 su **ON** per impostare l'indirizzo 6 dell'azionamento. Come impostazione predefinita, come mostrato nell'illustrazione precedente, tutti i microinterruttori di S1 e di S2 sono impostati su **ON** ad eccezione del primo microinterruttore di S1, che imposta l'indirizzo 127.

Impostare S3.2 su **ON** per attivare il resistore di terminazione.

14.4 Regolazione dell'IclA

Scopo di questa sezione

Questa sezione offre un esempio di regolazione dell'**IclA** con IclA Easy.

Contenuto di questa sezione

Questa sezione contiene le seguenti sottosezioni:

Argomento	Pagina
Configurazione dell'IclA in IclA Easy	191
Regolazione dell'IclA con IclA Easy	194

Configurazione dell'IclA in IclA Easy

Panoramica

Con IclA Easy, gli utenti possono definire le basi delle apparecchiature installate, descrivere le configurazioni associate e le impostazioni di comunicazione.

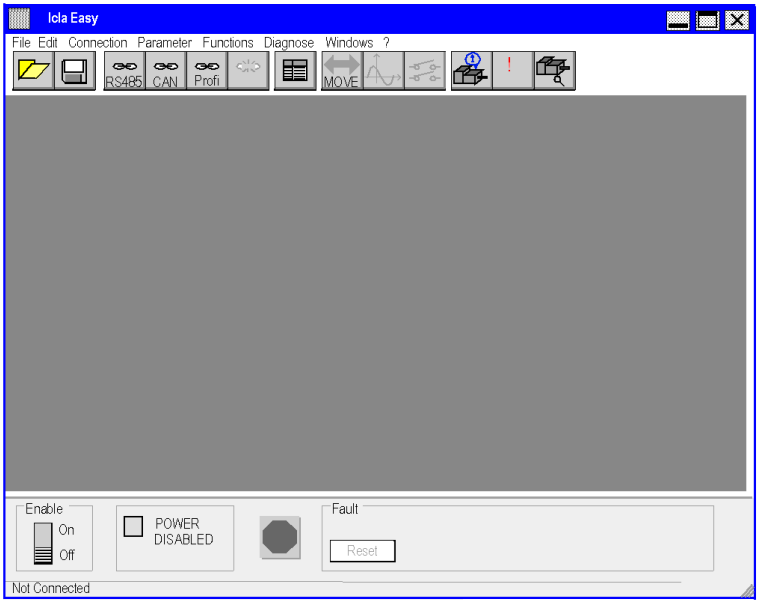
IclA Easy offre poi l'accesso a un gruppo di operazioni per la modifica o il trasferimento delle configurazioni nonché per la connessione alle apparecchiature.

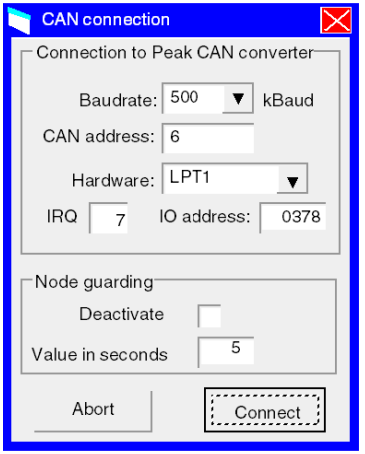
Il principio di navigazione di IclA Easy permette di associare un'interfaccia di configurazione con ogni tipo di apparecchiatura rendendone possibile il controllo, la regolazione e il monitoraggio.

NOTA: i segnali necessari, ad es. LIMN, LIMP, REF devono essere collegati o disattivati dal software di regolazione.

Connessione a IclA

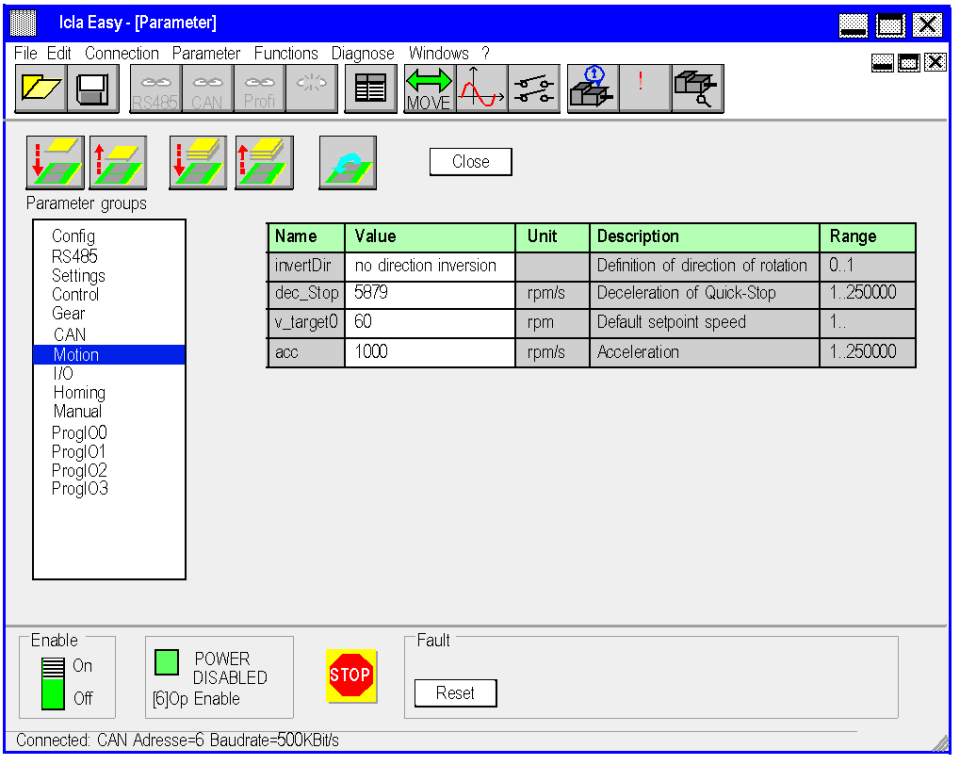
Questa tabella descrive la procedura di connessione a **IclA**:

Passo	Azione
1	Collegare il PC, su cui è installato IclA Easy per ATV71 al connettore PS/2 del Dongle PCAN sul servoazionamento da configurare.
2	Avviare IclA Easy per **IclA**. **Risultato:** viene visualizzata la schermata di avvio: 

Passo	Azione
3	Scegliere il comando **Connessione →Connessione CAN**. **Risultato:** viene visualizzata una casella di testo. 
4	La **Baudrate** deve essere impostata su 500 Kbaud. L'indirizzo **CAN** deve essere impostato su 6. L'**Hardware** deve essere impostato su LPT1 (Dongle PCAN PS/2). **Risultato:** viene avviato un trasferimento dati dal servoazionamento alla stazione di lavoro collegata.

Configurazione di base di IclA

L'esempio di seguito illustra la modifica del valore di accelerazione. Questa tabella descrive la procedura d'immissione di questa impostazione:

Passo	Azione
1	Dopo la connessione e il trasferimento delle configurazioni delle apparecchiature, IclA Easy visualizza una schermata di configurazione in una nuova finestra che offre l'accesso al controllo dell'apparecchiatura e alle funzioni di regolazione e monitoraggio.
2	Scegliere il parametro **Motion** nel **Gruppi parametri**. **Risultato:** viene visualizzata la finestra **Parametro**. 
3	Nella riga **acc**, l'accelerazione può essere impostata su 1000.
4	Salvare le impostazioni CANopen nella EEprom con il comando **Parametro →Invia gruppo parametri ad azionamento**. **Nota:** è possibile regolare le impostazioni del servoazionamento con la stessa procedura.
5	Una volta regolate le impostazioni, usare il comando **File →Chiudi** per scollegarsi.

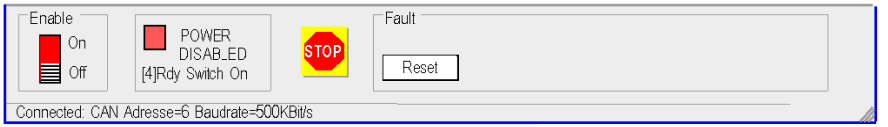
Regolazione dell'IclA con IclA Easy

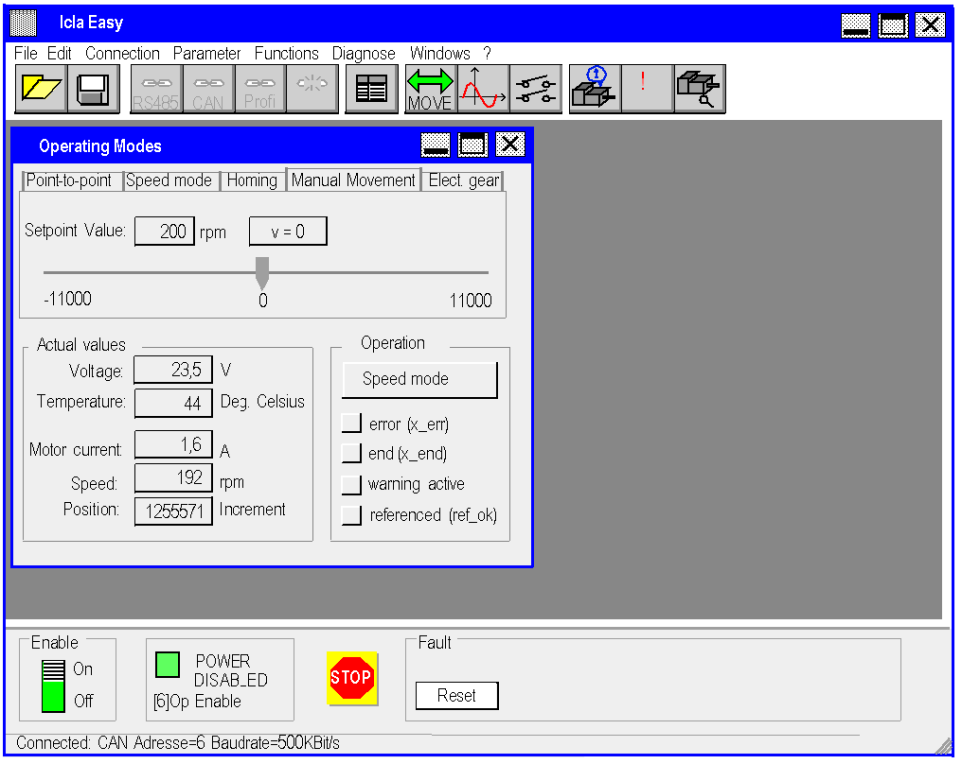
Preliminarmente

Si raccomanda di regolare la cinematica dell'asse prima che il programma lo avvii automaticamente.

Esempio di regolazione

La seguente tabella offre un esempio di regolazione della cinematica:

Passo	Azione
1	Collegarsi *(vedi pagina 191)* all'**IclA**.
2	La seguente figura mostra parte della nuova finestra. Questa finestra più in basso permette d'accedere alle funzioni di comando dell'**IclA**: 
3	Fare clic sul pulsante **Reset** per azzerare tutti i problemi.
4	Posizionare il cursore della zona **Enable** su **ON**.
5	Scegliere il comando **Funzioni** →**Modalità di funzionamento**. **Risultato:** viene visualizzata la finestra delle modalità di funzionamento.

Passo	Azione
6	Selezionare la scheda **Modalità di velocità** Immettere il valore 200 nella zona **Valore setpoint**. **Risultato:** il motore funziona e la sottofinestra è animata: 
7	Posizionare il cursore della zona **Attiva** su **OFF** una volta che la regolazione è terminata.

Indice analitico

B

blocchi funzione di movimento, *13*
Blocchi funzione di movimento
    ATV 31, *135*
    ATV 71, *165*
    avvio rapido, *13*
    IclA, *181*
    Lexium 05, *21*
    Lexium 32, *93*
Blocchi funzione di movimento (Motion)
    ATV 32, *151*
Blocchi funzioni di movimento
    Lexium 15LP/MP/HP, *111*
    metodologia, *19*

C

configurazione applicazione
    ATV 31, *135*
    Lexium 05, *21*
Configurazione dei servoazionamenti
    ATV, *159*
    ATV 31, *143*
    ATV 71, *173*
    IclA, *189*
    Lexium 05, *48*
    Lexium 15LP/MP/HP, *120*
    Lexium 32, *102*
Configurazione del bus CANopen, *29*
Configurazione dell'applicazione
    ATV 32, *151*
configurazione dell'applicazione
    ATV 71, *165*
    IclA, *181*
    Lexium 15LP/MP/HP, *111*
Configurazione dell'applicazione
    Lexium 32, *93*
Configurazione dell'asse, *36*

D

Debug dell'applicazione, *71*

O

Oscilloscopio, *107*

P

Programmazione dell'applicazione, *55*

R

Regolazione dei servoazionamenti
    ATV 31, *149*
    ATV 71, *179*
    IclA, *190*
    Lexium 05, *72*
    Lexium 15LP/MP/HP, *131*
    Lexium 32, *105*
Ricette, *79*

S

Sostituzione dei servoazionamenti, *84*